AIR TRAFFIC FLOW MANAGEMENT PROBLEM WITH STOCHASTIC CAPACITIES

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

INDUSTRIAL ENGINEERING

By

Efe Sertkaya

September 2021

Air Traffic Flow Management Problem with Stochastic Capacities
By Efe Sertkaya
September 2021

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Özlem Çavuş Iyigün(Advisor)

Güglielmo Lulli(Co-Advisor)

Ayşe Selin Kocaman

Sinan Güvel

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

AIR TRAFFIC FLOW MANAGEMENT PROBLEM WITH STOCHASTIC CAPACITIES

Efe Sertkaya
M.S. in Industrial Engineering
Advisor: Özlem Çavuş İyigün
Co-Advisor: Guglielmo Lulli
September 2021

Air traffic systems have substantial effects on transportation, logistics, and economics in a global scope. Due to both practical significance and intellectual challenges, air traffic flow management problems have been extensively studied for many decades. The aim of air traffic flow management problems is to plan the flow throughout the air traffic network while satisfying capacity constraints. In this study, we consider the case of stochastic capacities in the air traffic network. We propose both stochastic multistage integer and stochastic two-stage integer modeling approaches for the problem. In multistage and two-stage models, we aim to resolve the demand-capacity imbalances at each element of the air traffic network. To achieve this, we decide on the take-off times and routes of each flight for a given time horizon. We propose integer L-shaped and partial Benders' decomposition approaches to solve the two-stage model. Additionally, we analyze the effect of conditional value-at-risk constraints on delay time distributions. To incorporate conditional value-at-risk to solution methodologies, we propose a novel approximation technique. We present a detailed analysis of delay distributions, demonstrate the effect of the approximation technique on solution quality and computational performance. For computational experiments, we explicitly describe data generation procedures to obtain realistic instances. We demonstrate that the Partial Benders' modification outperforms the commercial solver (CPLEX) in almost every instance.

Keywords: Air traffic flow management, Two-stage stochastic programming, Integer L-shaped algorithm, partial Benders' decomposition, Conditional Value at Risk.

ÖZET

STOKASTİK KAPASİTELERLE HAVA TRAFİĞİ AKIŞ PROBLEMLERİ

Efe Sertkaya
Endüstri Mühendisliği, Yüksek Lisans
Tez Danışmanı: Özlem Çavuş İyigün
İkinci Tez Danışmanı: Guglielmo Lulli
Eylül 2021

Hava trafiği sistemleri ulaşım, lojistik ve ekonomiye dünya çapında önemli katkılar sağlamaktadır. Hem pratikteki önemi hem de sunduğu entelektüel zorluklar ile, hava trafiği akış problemleri geçtiğimiz birkaç on yılda çok fazla çalışılan bir konu olmuştur. Hava trafiği akış problemlerinin amacı, hava trafiği ağındaki akışı kapasite kısıtlarını göz önünde bulundurarak planlamaktır. Bu çalışmada, hava trafiği ağındaki kapasitelerin stokastik olduğu durum incelenmiştir. Problem için hem stokastik çok aşamalı tamsayı hem de stokastik iki aşamalı tamsayı modelleme yaklaşımları önerilmiştir. Hem çok aşamalı hem de iki aşamalı modellerde, hava trafik ağının her bir unsurundaki talep-kapasite dengesizliklerinin çözülmesi hedeflenmektedir. Bunu sağlamak için, belirli bir zaman dilimi içerisinde her uçuşun kalkış zamanlarına ve rotalarına karar verilmektedir. İki aşamalı modeli çözmek için tamsayılı L-şekilli ve kısmi Benders ayrıştırma yöntemleri önerilmiştir. Ek olarak, koşullu riske maruz değer kısıtlamalarının rotar süresi dağılımları üzerindeki etkisi analiz edilip, koşullu riske maruz değer kısıtlamalarını çözüm metodolojilerine dahil etmek için yeni bir yaklaşım tekniği önerilmiştir. Gecikme dağılımlarının ayrıntılı bir analizini sunarak, yaklaşım tekniğinin çözüm kalitesi ve hesaplama performansı üzerindeki etkisi incelenmiştir. Aynı zamanda hesaplamalı deneylerde gerçekçi örnekler elde etmek için, veri oluşturma prosedürlerinin detaylı bir açıklaması sunulmuştur. Kısmi Benders yaklaşımı'nın neredeyse tüm deneylerde, ticari amaçlı çözücü olan CPLEX'ten daha iyi performans gösterdiği gözlenmiştir.

Anahtar sözcükler: Hava trafiği akış yönetimi, Stokastik iki aşamalı programlama, Tamsayılı L-şekilli yöntem, Kısmi Benders' ayrıştırma, Koşullu riske maruz değer.

Acknowledgement

I would like to express my gratitude to a number of people who have contributed to me in many different ways.

First of all, I would like to thank my mother Beyhan, my father Muzaffer and my brother Ege for making the most important contributions to myself and supporting me no matter what. Most importantly, thank you for being a family to me.

I would like to express my utmost gratitude for my advisor Assoc. Prof. Özlem Çavuş İyigün for her endless patience and believing in me in the first place. She has successfully guided me throughout this journey and taught me many things along the way. She has become a very incredible first mentor for me at a job that I will hopefully do for my whole life.

I would like to thank Prof. Guglielmo Lulli for his contributions to this work. He is surely an expert in his domain and it is a great pleasure to work with him.

I would like to thank Asst. Prof. Çağın Ararat for delivering truly incredible lectures on probability theory, stochastic processes, financial engineering and reliability. His lectures changed the way I perceive the world and inspired me in many different levels.

I would like to express my thanks to Asst. Prof. Ayşe Selin Kocaman and Prof. Sinan Gürel for their time to read and review this thesis.

I hate to single out friends but there are some limitations on space. So, here it goes:

I would like to thank Utku Çallak for being my first and best friend, and thank you for making the phrase "brother from another mother" real for me.

Bilkent University has given me a lot, but perhaps the best it has given me is

a lifelong friend and colleague Emre Düzoylum. It was a pleasure and honor to be with him starting from the first day of my undergraduate studies.

I would like to thank Furkan Özden for being an inspiring friend. Although his combinatorial wish from our high school yearbook did not come true, I know that it still remains true in some sense.

I would like to thank Şifanur Çelik and İsmail Burak Taş for disproving that you can't make good friends after some age. I would like to thank İsmail Burak Taş for the incredible amount of help handling thesis related operations.

I would like to thank all the people who were a member of EA327 and EA325 offices during 2019-2020 academic year. Thank you for making graduate student offices fun!

Even though she explicitly asked me not to, I would also like to present my thanks anonymously to a person who was there for me in the last year of my Masters'.

Lastly, I would like to thank Bilkent University Industrial Engineering Department for giving me a path to follow with passion.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Air transportation systems include many complex processes that result in large amount of costs. Unsurprisingly, the air transportation network affects both local and global economies significantly. Air Transport Action Group [2] reports that the air transportation industry supported 65.5 million jobs and generated 3.5% of global GDP in 2017. In 2017, there were around 41.9 million scheduled commercial flights operated by 1,303 commercial airlines. The average number of flights scheduled per day globally is 120,000, which allowed 12 million passengers to travel worldwide in one day. With this enormous economic impact, management of growing air traffic has become increasingly significant over the years.

In both US and Europe, air transportation systems are managed by centralized collaborative decision-making authorities. European Union (EU) initiated a continent-wide project named Single European Sky (SES) in 2004 [3]. The objective of SES is to transform Air Traffic Management(ATM) from an intergovernmental practice to an EU practice. In Europe, EUROCONTROL [4] provides assistance to the unified strategic management of aviation operations since 1963 and assists many stakeholders with collaborative strategic decisions.

In the US, Federal Aviation Administration (FAA) manages daily air traffic consisting of 45,000 flights and 2.9 million airline passengers [5]. FAA acts as

a centralized strategic decision-making authority in National Airspace System (NAS) which is a network including air navigation facilities, airports and runways, information and services, rules and regulations, procedures and technical information etc.

With the increasing volume and economic significance of air traffic, ATM has become a substantial area of research in operations research and management science. ATM refers to quantitative and managerial studies about a broad collection of operations in aviation including air traffic flow management (ATFM), scheduling of airport runway operations, planning and control of aircraft trajectories. In the past few decades, various types of operations in aviation are modeled and enhanced using operations research techniques. As the demand for air transportation increases, modeling inherent stochasticity in air traffic has become more and more necessary to avoid congestions and delays.

ATFM considers en-route sectors and airports and aims to manage the flow of air traffic in a way that takes delays and operation costs into account. One of the main objectives of ATFM is to resolve demand-capacity imbalances in the air traffic domain. A demand-capacity imbalance occurs when the demand exceeds available capacity and it is mandatory to implement some control actions to equate demand and capacity. Usually, the possibility of modifying the capacity of the air traffic system is not available for an ATFM problem. Hence, capacities of airports and en-route sectors are exogenous parameters of the problem; only some flow control actions related to flights are available. Examples of available actions can be given as take-off times, routes etc.

Figure 1.1 displays the evolution of the amount of delay attributed to different causes in the air traffic system. Observe that, a significant portion of delays is attributed to Air Traffic Controller (ATC) capacity and weather disruptions. Hence, models concerning flight take-off decisions should consider the effect of these factors for the sake of practicality.

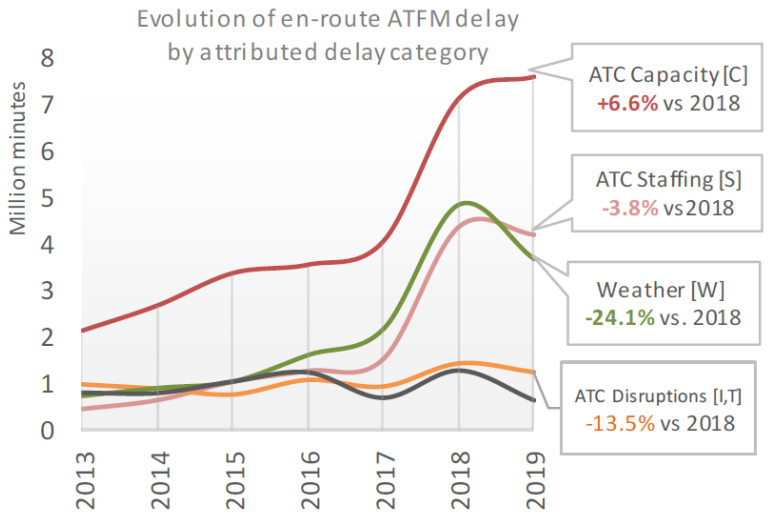


Figure 1.1: Evolution of causes of delays reported by EUROCONTROL in 2019 [1]

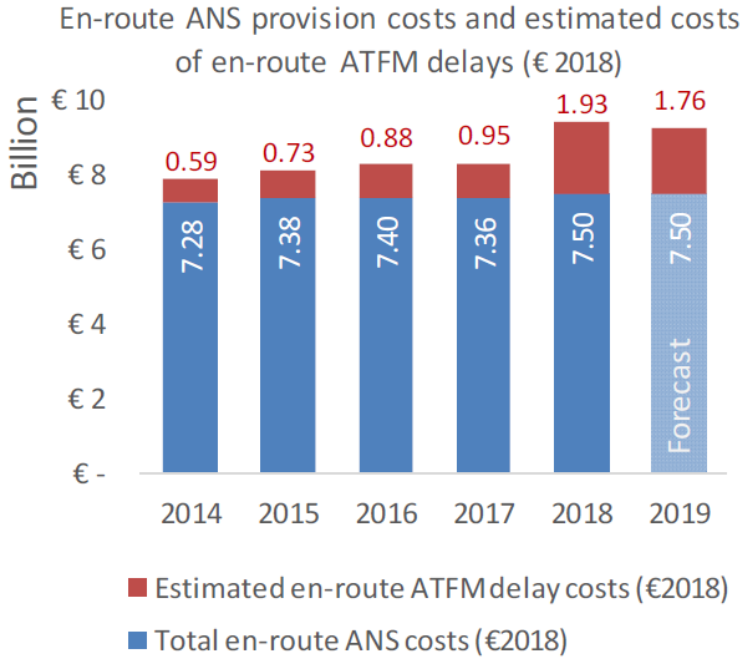


Figure 1.2: Evolution of costs related to delays and costs related to Air Navigation Services (ANS) reported by EUROCONTROL in 2019 [1]

This study defines Stochastic ATFM (SATFM) problem on a finite time horizon considering stochastic airport and en-route sector capacities to minimize delay and operation costs. Both of these cost components play a crucial role in Air Traffic Systems. In Figure 1.2, operation and delay costs of Europe's air traffic for years 2014 to 2019 are illustrated and it can be seen that billions of euros are spent on operating this system. So, any improvement can be translated into millions of euros easily.

SATFM aims to allocate the capacity reduction effects of flights to corresponding airspace elements such that operation and delay costs are minimized. Since airspace elements have stochastic capacities, we make decisions on take-off times and routes for each flight at different realizations of uncertain capacities. So that, the solution provides a prescription on flight take-off times and routes for each different scenario. We identify two types of SATFM models as two-stage and multistage SATFM. Differentiating factor for SATFM models is the structure of underlying stochasticity.

Practically, the two-stage model represents a system where capacities are known with certainty up to a point in the future. Hence, the resolution of uncertainty can be interpreted as a weather forecast being realized in a future time. Whereas the multistage model represents a system where uncertainty is resolved gradually. In the multistage model, we make decisions and observe the uncertain capacities more than once i.e. some decisions are made at different levels of uncertainty resolution. In Figure 1.3, we illustrate the relationship between observations of uncertainty and decisions for both types of SATFM.

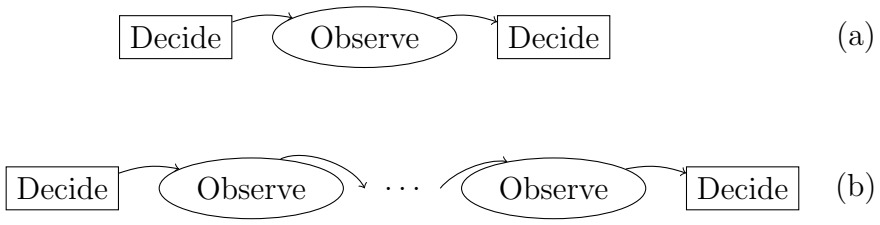


Figure 1.3: Decision making structures of two-stage (a) and multistage (b) models

In this study, we propose both multistage and two-stage formulations for

SATFM problem and provide solution methodologies for two-stage SATFM. Both models aim to find the optimal take-off time and route combination for each flight using the information given by different stages of capacities. For the two-stage model, we introduce Conditional Value-at-Risk (CVaR) constraints over delays. Since CVaR is a risk measure which quantifies the expected cost over a threshold, these constraints allow the control of variation beyond a point. Hence, CVaR constraints serve as a stable tool for controlling distributions of delays.

To solve the two-stage ATFM problem, we use two different algorithms. Both of these algorithms are based on the widely known Integer L-shaped method [6]. The first algorithm that we use to solve SATFM, is proposed by Angulo, Ahmed and Dey [7] and we refer to this algorithm as Improved Integer L-shaped. This algorithm provides enhancements on the Integer L-shaped method by providing alternative cut generating procedures. The second algorithm we use, is a combination of Improved Integer L-shaped and the method proposed by Crainic et al. [8]. This algorithm uses the partial decomposition approach proposed in [8] and uses the cut generation procedures from Improved Integer L-shaped. In Chapter 4, we provide two algorithms for the two-stage SATFM in detail and identify problem-specific structures. We also propose a novel approximation scheme for the CVaR-constrained two-stage problem using the combined methodology.

The organization of this thesis is as follows. In Chapter 2, we review the literature on Stochastic ATFM. In Chapter 3, we present multistage stochastic ATFM formulations and discuss the two-stage model based on them. We also elaborate on CVaR constraints on delays and discuss practical interpretations. In Chapter 4, we describe two Integer L-shaped method based solution methodologies for the two-stage problem and explicitly define problem-specific structures required for presented algorithms. Also, we propose a novel approximation method for the CVaR-constrained two-stage problem and present related proofs. In Chapter 5, we thoroughly describe realistic data generation methods and illustrate generated data. In Chapter 6, we present computational experiments including Value of Stochastic Solution (VSS) analyses and comparisons of different solution methodologies for varying parameters of the problem.

Chapter 2

Literature Review

ATFM problems attracted great interest from the scientific community due to the intellectual challenge posed by this class of problems as well as for the high potential impact of the research activities in this application domain. So far, great part of the ATFM related literature focused on deterministic mathematical models. However, due to the advancement of computational capabilities on both the hardware and software side, there has been a growing interest and emphasis in explicitly taking uncertainty into account in more recent time.

ATFM problems are first modeled in pioneering study of Odoni [9] where the author describes a stochastic, dynamic decision-making environment and decisions are constrained considering available capacity in an air traffic system. In [10], authors provided a state-of-the-art literature review focusing on deterministic and stochastic models and current practices in ATFM. They presented a classification based on practical purposes of different models. Most recently, [11] has provided a broad literature review focusing on all stochastic modeling applications in ATM including queueing, stochastic optimization and stochastic control practices.

Deterministic ATFM models have evolved to capture many aspects of air traffic systems. We mention few of the many deterministic models as foundations of

almost all the stochastic models in the literature. Bertsimas and Stock Patterson [12] proposed a strong integer programming (IP) formulation considering rerouting decisions and en-route sector capacities. They showed that in many cases solution of the linear programming (LP) relaxation of their formulation coincides with the IP solution. They also analyzed the polyhedral structure of the problem and show that under some specific assumptions the constraint matrix is totally unimodular. They define decision variables concerning traffic flows in a distinct manner incorporating routes implicitly as follows:

$$x_{st}^{f} = \begin{cases} 1 & \text{if flight } f \text{ arrives at sector } s \text{ by time } t \\ 0 & \text{otherwise} \end{cases}$$

Lulli and Odoni [13] proposed a deterministic ATFM model capturing Europe's air traffic. They have discovered many important charasteristics of solutions concerning collaborative decision making and elaborated on the trade-off between fairness and efficiency. They modeled the flow associated to a flight differently than Bertsimas and Stock Patterson [12] using take-off and landing times by defining the following decision variables:

$$x_{t}^{f} = \begin{cases} 1 & \text{if flight } f \text{ takes off at time } t \\ 0 & \text{otherwise} \end{cases}$$

$$y_{t}^{f} = \begin{cases} 1 & \text{if flight } f \text{ lands at time } t \\ 0 & \text{otherwise} \end{cases}$$

Bertsimas, Lulli and Odoni [14] proposed an IP formulation for large-scale deterministic ATFM problems and introduced three classes of valid inequalities utilizing special structures in the air traffic network. They also showed that the model is practically applicable in US air traffic due to short times experienced in computations.

Agustin et al. [15] proposed a deterministic formulation including flight cancellation, rerouting using a network structure. They provided significant computational insights about their formulations. They observed that solutions were

found without branching in many cases i.e. branch and cut tree consists of only one node.

Most of the contributions with stochasticity considerations have focused on the problem of constructing ground delay programs for aircrafts with stochastic airport capacity, namely, stochastic ground-holding problem. As an important distinction, ground-holding problems do not consider en-route sector capacities, they only focus on airports' arrival and departure capacities. Stochastic ground-holding problem is analyzed in relatively more detail for the single airport case (SAGHP).

Andreatta and Romanin-Jacur [16] focused on the single-period static case of SAGHP, developing a ground holding strategy to minimize the expected total delay, i.e., the sum of airborne holding and ground holding. The first multi-stage stochastic integer program with recourse for the SAGHP was proposed by Richetta and Odoni [17]. To mitigate the complexity of the problem, they developed an aggregate model whose focus (control actions) is on group of flights rather than individual flights. A relevant feature of the proposed model is that it can handle arbitrary cost functions of the ground delay assigned to flights; including non-negative, homogeneous and marginally non-decreasing cost functions that are of practical relevance for ground holding programs.

Under the assumption of linear ground delay costs, Ball et al. [18] provided an alternative aggregate formulation for the stochastic SAGHP. The very appealing feature of this formulation is that the constraint matrix enjoys a dual network structure, which guarantees that problem is polynomially solvable. As evidenced by Kotnyek and Odoni [19], the Richetta-Odoni model guarantees integer solutions under the following assumptions: *i*) every flight is considered in one cost category, and *ii*) the cost function is monotonically increasing with respect to ground delay. Monotonically increasing objective functions are particularly interesting since they result in prioritization according to flight schedules. This is in-line with the ration-by-schedule (RBS) allocation, which is commonly used practice.

A dynamic framework based on Richetta-Odoni model was proposed by Mukherjee and Hansen [20]. They provided a multi-stage IP formulation for the SAGHP that allows revisions on ground delay decisions based on updated capacity information.

There are very few contributions, especially concerning the problem that considers capacitation of both at airports and en-route sectors. Vranas, Bertsimas and Odoni [21] have proposed a multi-airport version of the stochastic ground-holding problem. More recently, Andreatta, Dell'Olmo, and Lulli [22] presented an aggregated stochastic model for the multi-airport capacity allocation problem to distribute capacities in ATFM.

To the best of our knowledge, the first model for the SATFM as mentioned before, although limited to only one route for a flight, was proposed by Alonso, Escudero and Ortuno [23]. The authors extended state-of-the-art Bertsimas-Stock Patterson [12] model to the stochastic setting and applied the fix-and-relax heuristic method to solve instances of the problem. They proposed a stochastic programming formulation where the uncertain parameters of the problem are represented by a discrete random vector.

Agustin et al. [24] provided a stochastic model that models flight rerouting to alternative routes in order to control congestion in en-route sectors along the original routes. Computational results for up to 400 flights were reported by the authors.

Bertsimas and Gupta [25] proposed the ATFM problem's first robust and adaptive formulation. They handled a stochastic version of the ATFM problem, which included weather-related uncertainty in airport and sector capacity, as well as adaptability and robustness. Based on a small number of bad weather occurrences travelling across different regions of the U.S. National Airspace System (NAS), the authors provide an uncertainty set for robust optimization. The authors presented and verified an equivalence relationship between the robust problem and a new deterministic instance. Based on this relationship, they proposed efficient solution methodologies. Real-world aircraft schedules and simulated weather fronts

were used to demonstrate. In the case of the robust problem, instances take into account no more than 1,000 flights. The authors noted that the robust problem exhibits some of the traits of the deterministic problem, and that the cost of robustness is often low.

Nilim and Ghaoui [26] address the problem of dynamically routing an aircraft under uncertain weather using a Markov decision process. They model the affect of weather to capacities as a stationary Markov chain and construct a Markov decision process to minimize expected delays. Also, Nilim and Ghaoui [27] proposed a methodology, based on a robust dynamic programming algorithm. The algorithm introduces a relationship between robustness of the solutions and the transition matrix being stochastic.

Ichoua [28] recently provided an overview of a potential solution methodology for the ATFM problem with stochastic capacities. The author considered the stochastic version of Bertsimas, Lulli, and Odoni [14] ATFM model. A distributed variant of the Sample Average Approximation (SAA) algorithm is described to solve the problem.

Chang, Solak, Clarke and Johnson [29] proposed a two-stage stochastic programming model considering capacity uncertainty induced by weather conditions and provided a rolling horizon heuristic to solve three-stage extension. They consider only one sector and model the demand-capacity imbalance of whole ATFM network using that sector.

Corolli et al. [30] proposed a two-stage stochastic programming model and progressive binary heuristic considering rerouting decisions. More in particular, all the flights are classified in three possible groups according to the corresponding level of decision-making flexibility. In the first group are flights whose decisions are here-and-now, in the second group are flights whose decisions can be revised and finally in the third group are flights scheduled late in the time horizon and are included in the model to estimate the downstream effect of the here-and-now decisions. This heuristic takes advantage of the key observation that the solution of the deterministic equivalent problem with relaxed second-stage variables is

binary for most instances.

Balakrishnan and Chandran [31] proposed a path-formulation for a stochastic integer program of the multistage ATFM. To solve instances of the problem, they developed a rounding heuristic combined to a column generation approach to compute the linear relaxations that attain close to optimal solutions. Using the proposed approach, they solved realistic instances of the US air traffic with 17,500 flights and scenario trees with up to 15 scenarios.

Chen, Chen and Sun [32] proposed a polynomial approximation algorithm utilizing chance-constrained programming for controlling the probability of violation on stochastic capacity constraints. In particular, they considered probability of deficient capacity for both airports and en-route sectors. Proposed solution methodology is a bisection algorithm which is a variant of supporting hyperplane method widely used to solve chance-constrained programs.

Solving stochastic optimization models were shown to be beneficial using Value of Stochastic Solution (VSS) analyses. Chang, Solak, Clarke and Johnson [29] demonstrated that VSS is between 6% and 30% increasing with the number of flights. Corolli et al. [30] observed that VSS values change depending on times of decisions. They argued that with more decisions made in first-stage of the problem VSS increases valuing up to 14%.

In this study, both multistage and two-stage SATFM formulations considering en-route sector and airport capacities is proposed. Unlike almost all the models in SATFM domain, we define our decision variables similarly to Lulli and Odoni [13]. While defining the decision variables, we also consider possible routes and address related capacity usage implicitly using a similar structure to Corolli et al. [30]. Our model differs from similar models in terms of definitions of decision variables. We define them in a way that allows the model to select from predetermined routes at the time of take-off. We also propose a solution methodology for two-stage SATFM problem. Utilizing two-stage structure, we solve SATFM problem using a modified version of Integer L-shaped method. In addition, we discuss practical implications and computational properties of CVaR constraints on delays. To

the best of our knowledge, constraints involving conditional value-at-risk (CVaR) were never used in SATFM domain.

Chapter 3

Mathematical Formulations

Consider an air traffic system, consisting of sectors and airports where sectors are defined as usable enclosed spaces in the air, see Figure 3.1 for an illustration of a small air traffic system on a map. Air traffic controllers are responsible for controlling the traffic in each sector and can handle a limited number of flights at each time period. Additionally, weather limits the ability to travel as airspace elements are affected by changing conditions. Hence, due to safety and congestion concerns, sectors are capacitated for each given time period.

When a flight descends (ascends) to arrive (depart) at (from) an airport, the arrival (departure) airport must have adequate resources dedicated to handle this operation. Similar to sectors, handling arrival and departure operations are also subject to air traffic controller availability and weather condition variations. Hence, we can model this limitation by specifying arrival/departure capacities for every airport and for each time period.

Different weather conditions affect each airspace elements' capacity depending on the geographical location and time periods in which the weather condition is active. For each time period, the system evolves according to realized capacities and flight-related decisions. For a time period t, we decide if a flight f takes off at that time period. If flight f takes off at time period t using route r, then it

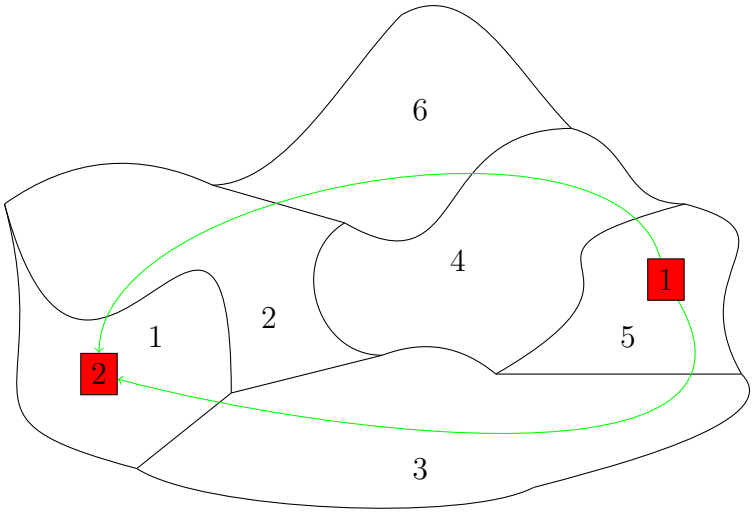


Figure 3.1: Illustration of an air traffic system on a map with 6 sectors and 2 airports (Routes from airport 1 to airport 2 are marked with green)

uses capacities of sectors included in route r. Route r determines which sectors flight f will pass through at each time period after the take-off.

In Figure 3.2, we illustrate two different routes for two different flights. For each flight, green arrows represent the first route available and red arrows represent the second route available. First flight is departing from airport 1 and arriving at airport 3. For this flight, first route goes through the sectors 2, 3 and 5 after 1, 2 and 3 time periods after departure time, respectively. Then it arrives at airport 3. If this flight takes off at time t using the first route, it uses one unit of capacity from capacities of sectors 2, 3 and 5 at times $t+1$, $t+2$ and $t+3$, respectively. It arrives (departs) at (from) airport 3 (airport 1) at time $t+4$ (t) and uses one unit of capacity from arrival (departure) capacity of airport 3 (airport 1) at time $t+4$ (t). Second flight is departing from airport 5 and arriving at airport 6. For this flight, first route goes through the sector 13 one time period after the departure time. Then it arrives at airport 6. If this flight takes off at time t using the first route, it uses one unit of capacity from capacity of sector 13 at time $t+1$. It arrives (departs) at (from) airport 5 (airport 6) at time $t+2$ (t) and uses one unit of capacity from arrival (departure) capacity of airport 6 (airport 5) at time $t+2$ (t).

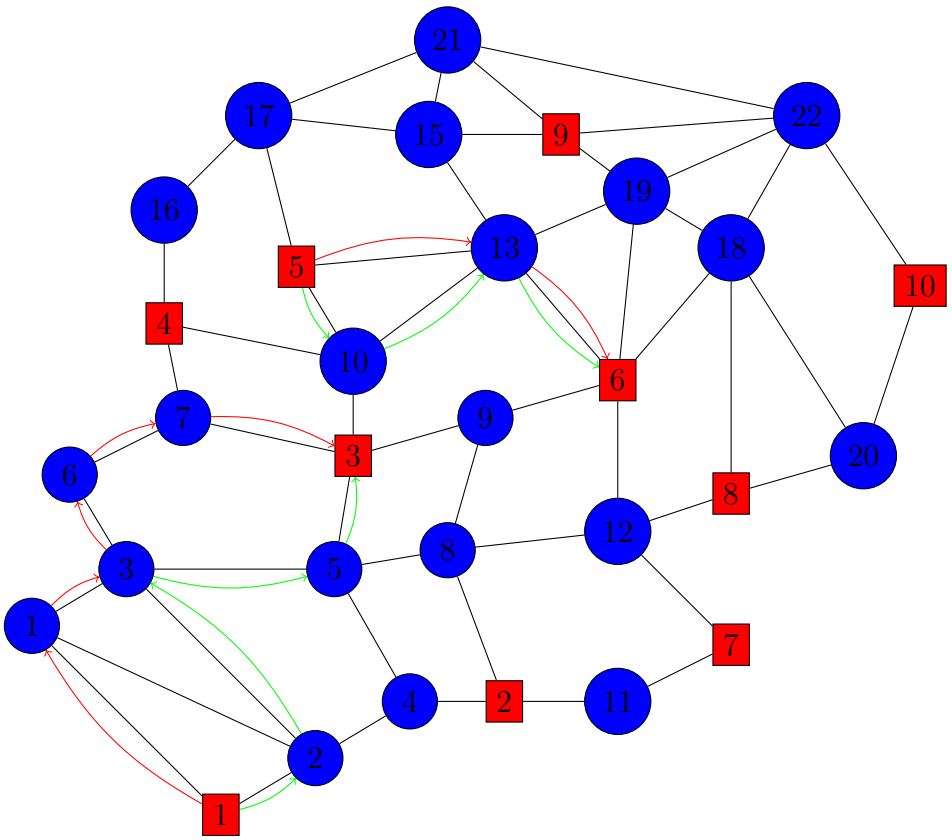


Figure 3.2: Examples of routes on an air traffic network with 10 airports (red) and 22 sectors (blue)

The Stochastic Air Traffic Flow Management (SATFM) problem is stated as follows. An airspace system formed by a set of airports and sectors, each subject to some capacity for every time period t over a set of time periods T is considered. Decisions are made with flight schedules. The flights use the airspace in the considered time horizon and are implicitly assigned with ground and airborne delays. Moreover, the route to go from origin to destination for any flight is selected according to the capacity constraints. Flight delays and routes are chosen with the objective of minimizing the summation of the cost of the resulting total delay and cost of traveling. The scope of the model is to model some of the possible flight control decisions. In particular, it makes decisions concerned with the entrance time of flights (departure time) into the air traffic system and their route to follow in the system. In this work, we do not consider the possibility of modifying the capacity of the air traffic system, i.e., the capacities of airports and en-route sectors are exogenous parameters of the problem, the only decisions available are those on the air traffic demand. In the planning phase of ATFM, to modify the demand released into the system, available control options are departure times or alternative routes to be assigned to flights.

Since capacities of different elements of airspace are not known with certainty before allocating flights to capacities of different elements, decision making is done as the uncertainty is resolved gradually. Hence, this problem can be modeled as a stochastic optimization model. However in practice, decisions can be revised anytime before the flight's departure and in some cases even during the execution of the flight. Therefore, the decision process is continuous rather than discrete. Therefore, the use of stochastic programming either two-stage or multistage is an approximation of the real decision process.

Structure of uncertainty is different in multistage and two-stage formulations which means that the process of realizing uncertain parameters differ, see Figure 1.3. In a two-stage setup, uncertainty is observed only once. This can be interpreted as the realization of a weather forecast scenario. Flight decisions related to first few time periods are considered as first stage decisions. So that, some flights may take off before the realization of uncertainty. Then, a scenario is realized and capacities of sectors and airports change accordingly. After the realization

of capacities, second stage flight decisions are made depending on the scenario.

In a multistage setup, uncertainty is observed more than once. This can be interpreted as consecutive realizations of weather scenarios. So that, first stage flight decisions are made before the first realization. After the observation, flight decisions until the second observation (second stage) are made depending on the capacities realized. After the second observation, decisions until the next observation (third stage) are made depending on the whole history of observations, and the decision and realization sequences follow similarly. Observations of uncertainty are made at the beginning of some time periods, i.e. some time periods can be identified as uncertainty resolution times. This implies that stages consist of multiple time periods. Since flight decisions in previous time periods affect the usage of capacities in future time periods, decisions of different stages are connected in terms of capacity usage.

While both two-stage and multistage approaches have a variety of solution methodologies, two-stage stochastic programming provides a modeling framework which is computationally much more efficient. As the uncertainty structure gets more complex, the computational effort to solve stochastic programs drastically increases. Since both formulations are approximations to the real-life decision process, we solve two-stage SATFM without loss of reality.

This chapter presents different formulations for SATFM problem including multistage and two-stage formulations. Although we provide both multistage and two-stage models, we present a solution methodology for solutions of two-stage model only. In some extreme weather scenarios, resulting delays tend to distribute unevenly among scenarios. To control delays against undesired distributions, we also present an alternative two-stage risk-averse model to control distribution of delays using CVaR constraints.

3.1 Multistage Formulation

In this section, we provide a multistage SATFM model. We introduce the following notation to formulate the SATFM with multiple stages.

Sets

F : set of flights

$\mathcal{A}$: set of airlines

$\bar{\mathcal{A}}_i$: set of flights operated by airline $i \in \mathcal{A}$

T : set of time periods

S : set of enroute sectors

A : set of airports

$\mathcal{C}$: set of pair of flights that are connected

R^f : set of feasible routes for flight $f \in F$

T^f : feasible set of take-off times for flight $f \in F$ i.e. $T^f := \{t \in T : t \geq \sigma^f\}$

Ω : set of scenarios

Ω^t_ω : set of scenarios that have the same history as scenario $\omega \in \Omega$ up to period $t \in T$

Parameters

e^{ft}_r : cost associated with route $r \in R^f$ if flight $f \in F$ takes off at time $t \in T$

t^f_{rk} : flight time for flight $f \in F$ to reach its destination $k \in A$ along route $r \in R^f$

t^f_{rs} : flight time for flight $f \in F$ to reach sector $s \in S$ along route $r \in R^f$

$A^t_{k\omega}$: capacity of arrival airport $k \in A$ at time $t \in T$ under scenario $\omega \in \Omega$

$D^t_{k\omega}$: capacity of departure airport $k \in A$ at time $t \in T$ under scenario $\omega \in \Omega$

$C^t_{s\omega}$: capacity of sector $s \in S$ at time $t \in T$ under scenario $\omega \in \Omega$

p_ω : probability of scenario $\omega \in \Omega$

a^f : arrival airport of flight $f \in F$

d^f : departure airport of flight $f \in F$

δ : turnaround time

τ^f : planned arrival time of flight $f \in F$

σ^f : scheduled departure time of flight $f \in F$

$\mathcal{M}$: maximum delay allowed for total airline delay

Note that, routes are defined as tuples of sector time pairs which indicate the time needs to be flown by the aircraft using the route to reach the corresponding sector. Formally, in (s_k, t_k), where s_k is the k^{th} sector flown along the route and t_k is the time of entering s_k. Two examples of routes can be given based on the system illustrated with Figure 3.1,

$$r_1 = [(5,4),(3,5),(1,7)], r_2 = [(5,1),(4,4),(6,7),(2,9),(1,10)].$$

If the aircraft uses the route r_1 given above and takes off at time 0, the aircraft will be at sector 5 at time 4, then it will be at sector 3 at time 5 and then it will be at sector 1 at time 7. If it uses the route r_2 given above and takes off at time 0, the aircraft will be at sector 5 at time 1, then it will be at sector 4 at time 4, then it will be at sector 6 at time 7, then it will be at sector 2 at time 9 and then it will be at sector 1 at time 10.

We use as criterion of performance the total amount of delays assigned for an airline which also includes reactionary delays. Reactionary delays are delays that are caused by the late arrival of aircraft, crew, passengers or loads from a previous journey. These delays usually spread over the air traffic network due to the aircraft, crew and passengers connections. Set of pair of flights that are connected by either aircraft, crew or passenger is denoted as $\mathcal{C}$. Turnaround time (δ) is the time needed for an aircraft to be ready for another flight after arrival. So that, if two flights are connected the subsequent flight cannot take off before the preceding flight's arrival.

Decision Variables

For $f \in F, t \in T^f, r \in R^f, \omega \in \Omega$, we define the following:

$$x^f_{r\omega}(t) = \begin{cases} 1 & \text{if flight } f \text{ takes off at time } t \text{ using route } r \text{ under scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

Mathematical Program (MSATFM)

$$\min \sum_{\omega\in\Omega} p_\omega \left(\sum_{f\in F} \sum_{r\in R^f} \sum_{t\in T^f} e_r^{ft} x_{r\omega}^f(t) \right) \tag{3.1}$$

$$\text{s.t.} \sum_{r\in R^f} \sum_{t\in T^f} x_{r\omega}^f(t) = 1, \qquad \forall f \in F, \omega \in \Omega \tag{3.2}$$

$$\sum_{f\in F: a^f = k} \sum_{r\in R^f} x_{r\omega}^f(t - t_{rk}^f) \le A_{k\omega}^t, \qquad \forall k \in A, t \in T, \omega \in \Omega \tag{3.3}$$

$$\sum_{f\in F: d^f = k} \sum_{r\in R^f} x_{r\omega}^f(t) \le D_{k\omega}^t, \qquad \forall k \in A, t \in T, \omega \in \Omega \tag{3.4}$$

$$\sum_{f\in F} \sum_{r\in R^f: s\in r} x_{r\omega}^f(t - t_{rs}^f) \le C_{s\omega}^t, \qquad \forall s \in S, t \in T, \omega \in \Omega \tag{3.5}$$

$$\sum_{r\in R^f} \sum_{\tau \ge t - t_{ra^f}^f - \delta} x_{r\omega}^f(\tau) + \sum_{r\in R^{f'}} \sum_{\tau \le t} x_{r\omega}^{f'}(\tau) \le 1 \qquad \forall (f, f') \in \mathcal{C}, t \in T, \omega \in \Omega \tag{3.6}$$

$$\sum_{f\in \bar{\mathcal{A}}_k} \sum_{r\in R^f} \sum_{t\in T^f} x_{r\omega}^f(t) \left[t + t_{ra^f}^f - \tau^f \right]_+ \le \mathcal{M}, \qquad \forall k \in \mathcal{A}, \omega \in \Omega \tag{3.7}$$

$$x_{r\omega}^f(t) = x_{r\omega'}^f(t), \qquad \forall f \in F, r \in R^f, t \in T, \quad \omega \in \Omega, \omega' \in \Omega_\omega^t \tag{3.8}$$

$$x_{r\omega}^f(t) \in \{0, 1\}, \qquad \forall f \in F, r \in R^f, t \in T, \quad \omega \in \Omega \tag{3.9}$$

In (MSATFM), objective function (3.1) minimizes expected total operation cost. Constraint (3.2) enforces each flight to take off using exactly one route-time combination in any scenario. Constraints (3.3) and (3.4) make sure the arrival and departure capacities, respectively, of each airport are not exceeded in any scenario. Constraint (3.5) makes sure that the capacity of sectors in the airspace is not exceeded for all time periods in any scenario. Constraint (3.6) makes sure that in any scenario and for any connected pair of flights, the latter flight does not take off before the arrival of the preceding flight plus a turnaround time. With constraint (3.6), one can effectively model the multiple usage of an aircraft or a crew for different flights. Constraint (3.7) makes sure that the total delay for an airline cannot exceed a given amount in any scenario. Constraint (3.8) is nonanticipativity constraint which imposes that the revealed information cannot be used in decision making before its time of reveal. Constraint (3.9) is domain constraint.

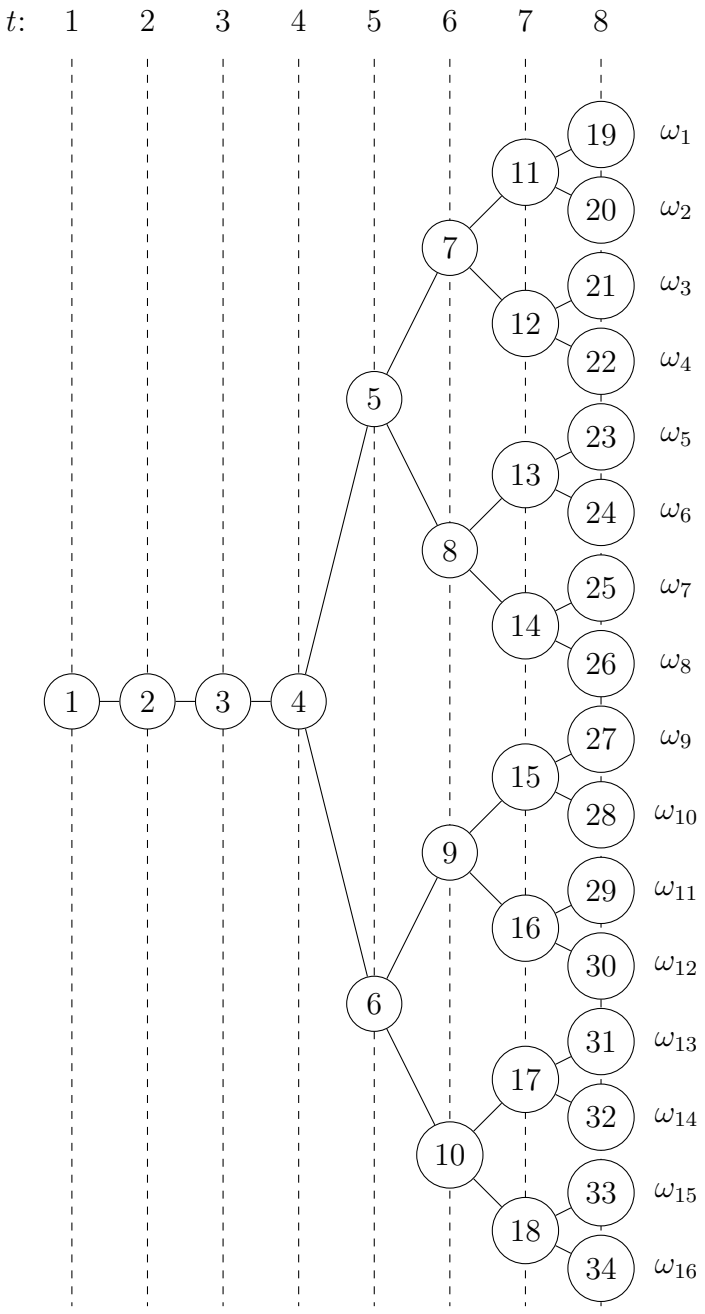


Figure 3.3: Example of a multistage scenario tree considering 8 time periods and 16 scenarios

In Figure 3.3, we illustrate the structure of uncertainty in (MSATFM) on a small example. In this example, we consider 16 scenarios where first stage is up to $t = 4$ and other stages are in correspondence with time indices. So that, uncertainty resolutions occur at $t = 4, 5, 6, 7$. In Table 3.1, we identify some aforementioned structures used for modeling nonanticipativity. For every time period, identified parameters associate a set of scenarios with each scenario and uniquely identify the common history structure implied by the scenario tree.

Parameter	Value
$\Omega^5_{\omega_1}$	$\{\omega_2, \omega_3, \omega_4, \omega_5, \omega_6, \omega_7, \omega_8\}$
$\Omega^6_{\omega_1}$	$\{\omega_2, \omega_3, \omega_4\}$
$\Omega^6_{\omega_{14}}$	$\{\omega_{13}, \omega_{15}, \omega_{16}\}$
$\Omega^7_{\omega_{14}}$	$\{\omega_{13}\}$

Table 3.1: Values of some parameters for the example scenario tree in Figure 3.3

3.2 Two-stage Formulations

In this section, we present the formulation of two-stage Stochastic Air Traffic Flow Management Problem and provide a CVaR-constrained version of the problem. We also discuss practical interpretations of CVaR constraints on airline delays.

3.2.1 Two-stage ATFM

In a multi-stage decision making environment, uncertainty is gradually resolved as stages proceed. In a two-stage decision making environment, some decisions are made before the uncertainty is resolved and some decisions are made after. The transition from multistage model to two-stage model can be interpreted as relaxing a subset of nonanticipativity constraints (3.8), i.e. after the first resolution of uncertainty we consider all the uncertainty is resolved.

Since many of the conditions in the system are certain for closer times, modeling the capacities of first few time periods as deterministic is practically accurate. This observation allows us to classify first few periods as first stage time periods. Let $\bar{t} \in T$, we relax the following nonanticipativity constraints in (MSATFM) to obtain a two-stage problem:

$$x^f_{r\omega}(t) = x^f_{r\omega'}(t), \forall f \in F, r \in R^f, t \in \{t \in T : t \geq \bar{t}\}, \omega \in \Omega, \omega' \in \Omega^t_\omega.$$

This implies that, for the first stage time periods, i.e. $t \leq \bar{t}$ and $\forall f \in F, r \in R^f$, the following is true:

$$x^f_r(t) = x^f_{r\omega}(t), \forall \omega \in \Omega.$$

In Figure 3.4, we illustrate the transition from a multistage environment to two-stage environment by relaxing nonanticipativity constraints. In the setting of Figure 3.3, we relax nonanticipativity constraints after $t = 4$ so that we create copies of nodes in the scenario tree corresponding to each scenario $\omega \in \Omega$. In this example, uncertainty is resolved only once after $t = 4$ allowing us to model the problem as a two-stage stochastic optimization problem. To construct a two-stage model we define the following in addition to the notation in (MSATFM):

Sets

T_1 : set of time periods in first stage

T_2 : set of time periods in second stage

Parameters

A^t_k : capacity of arrival airport $k \in A$ at time $t \in T_1$

D^t_k : capacity of departure airport $k \in A$ at time $t \in T_1$

C^t_s : capacity of sector $s \in S$ at time $t \in T_1$

$A^t_{k\omega}$: capacity of arrival airport $k \in A$ at time $t \in T_2$ under scenario $\omega \in \Omega$

$D^t_{k\omega}$: capacity of departure airport $k \in A$ at time $t \in T_2$ under scenario $\omega \in \Omega$

$C^t_{s\omega}$: capacity of sector $s \in S$ at time $t \in T_2$ under scenario $\omega \in \Omega$

We partition the set of time periods into two sets denoting first and second stage time periods. Since for $t \in T_1$ capacites of airports and sectors are constant, we define them independent of scenarios.

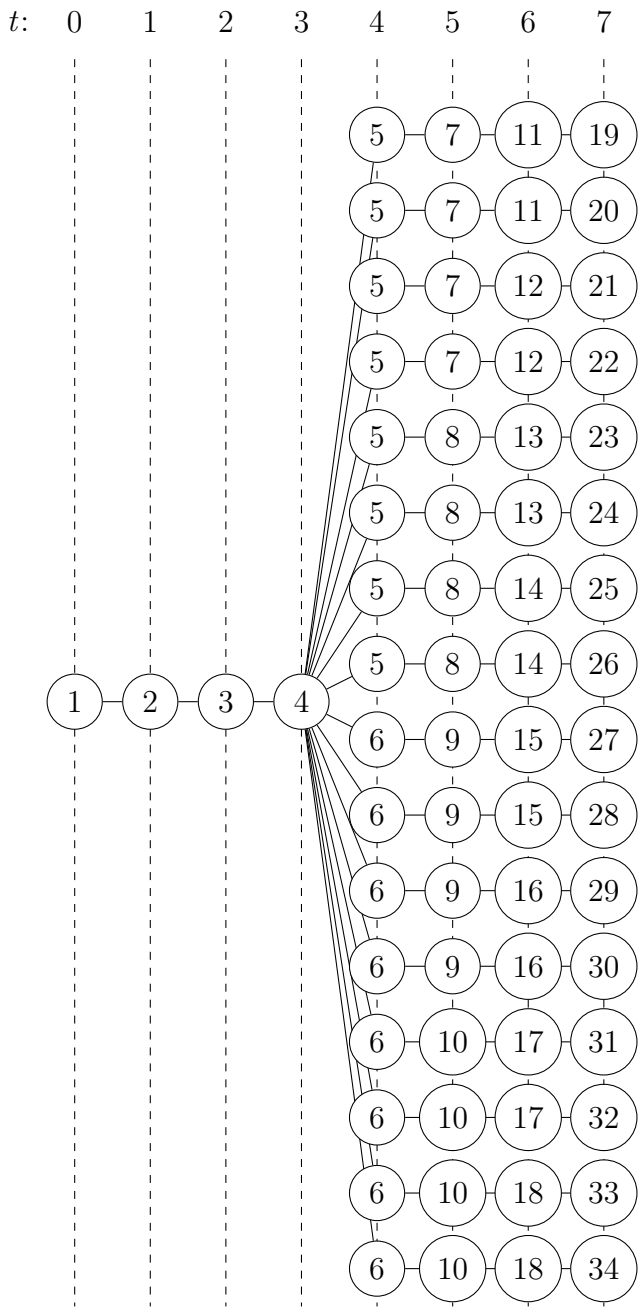


Figure 3.4: Two-stage relaxation of the scenario tree example given in Figure 3.3

We construct the model as follows:

Decision Variables

First Stage

For $f \in F, t \in T^f \cap T_1, r \in R^f$, we define the following:

$$x_r^f(t) = \begin{cases} 1 & \text{if flight } f \text{ takes off at time } t \text{ using route } r \\ 0 & \text{otherwise} \end{cases}$$

Second Stage

For $f \in F, t \in T^f \cap T_2, r \in R^f, \omega \in \Omega$, we define the following:

$$x_{r\omega}^f(t) = \begin{cases} 1 & \text{if flight } f \text{ takes off at time } t \text{ using route } r \text{ under scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

Mathematical Program (2SATFM)

$$\min \sum_{f\in F}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} e_r^{ft}x_r^f(t) + \sum_{\omega\in\Omega} p_\omega \left(\sum_{f\in F}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2} e_r^{ft}x_{r\omega}^f(t)\right) \tag{3.10}$$

$$\text{s.t.} \sum_{f\in F:a^f=k}\sum_{r\in R^f} x_r^f(t-t_{rk}^f) \le A_k^t, \quad \forall k \in A, t \in T_1 \tag{3.11}$$

$$\sum_{f\in F:d^f=k}\sum_{r\in R^f} x_r^f(t) \le D_k^t, \quad \forall k \in A, t \in T_1 \tag{3.12}$$

$$\sum_{f\in F}\sum_{r\in R^f:s\in r} x_r^f(t-t_{rs}^f) \le C_s^t, \quad \forall s \in S, t \in T_1 \tag{3.13}$$

$$\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} x_r^f(t) + \sum_{r\in R^f}\sum_{t\in T^f\cap T_2} x_{r\omega}^f(t) = 1, \quad \forall f \in F, \omega \in \Omega \tag{3.14}$$

$$\sum_{f\in F:a^f=k}\sum_{r\in R^f} \left(x_r^f(t-t_{rk}^f)1_{\{t-t_{rk}^f\in T_1\}} + x_{r\omega}^f(t-t_{rk}^f)1_{\{t-t_{rk}^f\in T_2\}}\right) \le A_{k\omega}^t,$$

$$\forall k \in A, t \in T_2, \omega \in \Omega \tag{3.15}$$

$$\sum_{f\in F:d^f=k}\sum_{r\in R^f} x_{r\omega}^f(t) \le D_{k\omega}^t, \quad \forall k \in A, t \in T_2, \omega \in \Omega \tag{3.16}$$

$$\sum_{f\in F}\sum_{r\in R^f:s\in r} \left(x_r^f(t-t_{rs}^f)1_{\{t-t_{rs}^f\in T_1\}} + x_{r\omega}^f(t-t_{rs}^f)1_{\{t-t_{rs}^f\in T_2\}}\right) \le C_{s\omega}^t,$$

$$\forall s \in S, t \in T_2, \omega \in \Omega \tag{3.17}$$

$$\sum_{r\in R^f}\sum_{\tau\ge t-t_{raf}^f-\delta} \left(x_r^{f'}(\tau)1_{\tau\in T_1} + x_{r\omega}^{f'}(\tau)1_{\tau\in T_2}\right)$$

$$+\sum_{r\in R^{f'}}\sum_{\tau\le t}\left(x_r^{f'}(\tau)1_{\tau\in T_1}+x_{r\omega}^{f'}(\tau)1_{\tau\in T_2}\right)\le 1,\forall(f,f')\in\mathcal{C},t\in T,\omega\in\Omega \tag{3.18}$$

$$\sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1}x_r^f(t)\left[t+t_{raf}^f-\tau^f\right]_+ + \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2}x_{r\omega}^f(t)\left[t+t_{raf}^f-\tau^f\right]_+\le\mathcal{M},\forall k\in\mathcal{A},\omega\in\Omega \tag{3.19}$$

$$x_{r\omega}^f(t)\in\{0,1\},\quad\forall f\in F,r\in R^f,t\in T^f\cap T_2,\omega\in\Omega \tag{3.20}$$

$$x_r^f(t)\in\{0,1\},\quad\forall f\in F,r\in R^f,t\in T^f\cap T_1 \tag{3.21}$$

In (2SATFM), objective function (3.10) minimizes expected total operation cost. Constraint (3.14) enforces each flight to take off using exactly one route-time combination in any scenario. Constraints (3.11) and (3.12) make sure the arrival and departure capacities, respectively, of each airport are not exceeded at first stage. Constraints (3.15) and (3.16) make sure the arrival and departure capacities, respectively, of each airport are not exceeded for any scenario at second stage. Constraints (3.13) and (3.17) make sure that the capacity of sectors in the airspace is not exceeded at both first and second stage periods for any scenario. Constraint (3.18) implies that in any scenario and for any connected pair of flights, the latter flight does not take off before the arrival of the preceding flight plus a turnaround time. With constraint (3.18), one can effectively model the multiple usage of an aircraft or a crew for different flights. Constraint (3.19) denotes that the total delay for an airline cannot exceed a given amount in any scenario. Constraints (3.21) and (3.20) are domain constraints for first and second stage variables.

3.2.2 Conditional Value-at-Risk (CVaR) Constraints for Delays

Total delay constraints for each airline (3.19) limit the delay by simply setting an upper bound for every scenario. Although this approach provides a limitation on delays, it fails to prevent delays from distributing unevenly among scenarios. Under extreme weather conditions, resulting delays tend to cluster around $\mathcal{M}$ when

maximum delay constraints are used. Using CVaR constraints allows control over clustered delay points and provides evenly distributed airline delays.

CVaR is a risk measure which quantifies expected loss exceeding a quantile value. For a random variable X representing loss, Conditional Value-at-Risk at α is defined as follows:

$$\mathrm{CVaR}_\alpha(X) = \inf_{\eta\in\mathbb{R}} \left\{ \eta + \frac{1}{1-\alpha}\mathbb{E}\big[(X-\eta)_+\big] \right\}.$$

By replacing the constraints (3.19) with risk constraints on total airline delays, it is possible to achieve less conservative solutions while controlling the delay levels. To model risk-aversion, we use CVaR as a risk measure which is a coherent measure of risk capturing various aspects of decision makers' risk aversion [33], [34].

To define CVaR constraints on airline delays for every airline $k \in \mathcal{A}$, we define the following random variable depending on first and second stage variables $(x_r^f(t))_{\omega\in\Omega, f\in F, r\in R^f, t\in T_1}$ and $(x_{r\omega}^f(t))_{\omega\in\Omega, f\in F, r\in R^f, t\in T_2}$:

$$Y_k(\omega) = \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} x_r^f(t)\big[t+t_{ra^f}^f-\tau^f\big]_+ + \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2} x_{r\omega}^f(t)\big[t+t_{ra^f}^f-\tau^f\big]_+.$$

Using $Y_k : \Omega \to \mathbb{R}$, we introduce the following constraint for each airline $k \in \mathcal{A}$:

$$\mathrm{CVaR}_\alpha\big(Y_k\big) \le \beta. \tag{3.22}$$

For random variables defined on a finite probability space, CVaR constraints can be conveniently put in a linear form.

Theorem 3.2.1 (Linearization of CVaR)**.** *Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a finite probability space and $X : \Omega \to \mathbb{R}$ be a random variable on $(\Omega, \mathcal{F}, \mathbb{P})$. Then,*

$$\begin{aligned}\mathrm{CVaR}_\alpha(X) \le \beta \iff & \eta + \frac{1}{1-\alpha}\sum_{w\in\Omega} p_\omega w_\omega \le \beta \\ & w_\omega \ge X(\omega) - \eta, \forall\omega\in\Omega \\ & w_\omega \ge 0, \forall\omega\in\Omega \\ & \eta\in\mathbb{R}\end{aligned}$$

Proof. We refer to Rockafellar and Ursayev [33] for the proof of this theorem. □

To use the linear form of CVaR, we introduce the following decision variables and use the notation from (2SATFM).

w_ω^k : auxillary variable for scenario $\omega \in \Omega$ and airline $k \in \mathcal{A}$

η^k : auxillary variable for airline $k \in \mathcal{A}$

Using Theorem 3.2.1 we can rewrite (3.22) as follows:

$$\begin{aligned}
\text{CVaR}_\alpha(Y_k) \leq \beta \iff & \eta^k + \frac{1}{1-\alpha}\sum_{w\in\Omega} p_\omega w_\omega^k \leq \beta \\
& w_\omega^k \geq \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} x_r^f(t)\left[t + t_{ra^f}^f - \tau^f\right]_+ \\
& \quad + \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2} x_{r\omega}^f(t)\left[t + t_{ra^f}^f - \tau^f\right]_+ - \eta^k, \forall\omega\in\Omega \\
& w_\omega^k \geq 0, \forall\omega\in\Omega \\
& \eta^k \in \mathbb{R}
\end{aligned}$$

Extensive formulation of CVaR constrained two-stage stochastic air traffic flow management problem is given by: (R2SATFM)

$$\begin{aligned}
&\min(3.10) \\
&\text{s.t.}(3.11)-(3.18),(3.20)-(3.21) \\
&\quad \eta^k + \frac{1}{1-\alpha}\sum_{w\in\Omega} p_\omega w_\omega^k \leq \beta, \quad \forall k\in\mathcal{A} && (3.23) \\
&\quad w_\omega^k \geq \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} x_r^f(t)\left[t + t_{ra^f}^f - \tau^f\right]_+ \\
&\quad + \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2} x_{r\omega}^f(t)\left[t + t_{ra^f}^f - \tau^f\right]_+ - \eta^k, \forall\omega\in\Omega, k\in\mathcal{A} && (3.24) \\
&\quad w_\omega^k \geq 0, \qquad \forall\omega\in\Omega, k\in\mathcal{A} && (3.25) \\
&\quad \eta^k \in \mathbb{R}, \qquad \forall k\in\mathcal{A} && (3.26)
\end{aligned}$$

In (R2SATFM), maximum delay constraints given by (3.19) are replaced with constraints (3.23)-(3.26) which are making sure that the Conditional Value-at-Risk on total delay for an airline cannot exceed a given value β. All other constraints are exactly the same with (2SATFM).

Chapter 4

Solution Methodology

In practice, flight control actions are rather continuous than discrete, i.e. real-time decisions can be made on flight take-off times and routes. However, uncertainty structures in air traffic systems can be more efficiently utilized using discrete settings, i.e. uncertain capacity observations can be efficiently approximated with discrete values. Hence, multistage and two-stage models provided in Chapter 3, are both approximations of real-life decision making process. Furthermore, since these models practically used to assist the current decisions considering the future, two-stage model provides a less restrictive structure on uncertainty compared to the multistage model. Thereby, it can usually be more applicable in real-life decision making for air traffic systems. In this chapter, we only focus on solution methodologies for the two-stage model.

Two-stage stochastic programs are inherently difficult to solve since problem sizes get exponentially larger with number of scenarios. Hence, methodological studies about stochastic programming have inherited many algorithms from large-scale optimization. Introducing binary and integer decision variables to stochastic programming problems, increases the computational difficulty. Integer programming (IP) problems are already hard to solve and stochasticity amplitudes the hardship by increasing the problem size.

There are many algorithms designated for solving two-stage stochastic programming problems based on the idea of Benders' Decomposition [35]. The idea to use this approach for two-stage stochastic linear programming problems by decomposing the problem with respect to scenarios, so called L-shaped algorithm, is proposed by Van Slyke and Wets [36]. L-shaped algorithm utilizes the structure induced by scenario dependency of the problem and applies Benders' Decomposition by considering scenario dependent variables in different subproblems and considering first stage variables in the master problem. Multi-cut L-shaped algorithm by Birge and Louveaux [37] is a variant of L-shaped algorithm using multiple optimality cuts in one major iteration, as opposed to a single optimality cut in one iteration of L-shaped algorithm. Authors demonstrate that multi-cut approach may lead to faster convergence to optimality in some cases.

For stochastic integer programs, the Integer L-shaped method proposed by Laporte and Louveaux [6] introduces a novel branch and cut procedure for solving two-stage stochastic integer programs by introducing an additional class of optimality cuts called Integer L-shaped cuts.

A significant portion of methodological studies in two-stage stochastic programming literature focused on algorithms that can be considered as variants of aforementioned methods. More recently, Angulo, Ahmed and Dey [7] proposed an enhancement procedure for the Integer L-shaped method by proposing two alternative cut adding procedures, which we describe in detail in Section 4.2 and we refer to this algorithm as Improved Integer L-shaped. Crainic et al. [8] described a Partial Benders' Decomposition approach and proposed a methodology to select existing scenarios or create artificial scenarios based on existing ones to include in the master problem to reduce the number of necessary iterations. To solve (2SATFM) problem defined in Section 4.4, we present a combination of Improved Integer L-shaped and Partial Benders'. In particular, we utilize an alternative cut generation method from Improved L-shaped and the scenario selection method from Partial Benders' to construct an efficient solution methodology for (2SATFM).

In this chapter, we first describe the L-shaped algorithm for linear two-stage

stochastic programs to serve as a basis in Section 4.1. After building a basis, we describe the Improved Integer L-shaped in detail in Section 4.2. After a detailed description of Improved Integer L-shaped, we describe the scenario selection method from Partial Benders' to modify the Improved Integer L-shaped. We also provide details of problem specific structures to use in algorithms for (2SATFM) problem. Lastly, in Section 4.5, we present a novel approximation scheme for (R2SATFM) problem using the combined methodology.

4.1 L-shaped Algorithm

In this section, we briefly describe the multi-cut L-shaped Algorithm proposed by Birge and Louveaux [37]. Since this method serves as a building block for subsequent methods provided, we use a common notation to describe all the algorithms.

The multi-cut L-shaped Algorithm is designed to solve optimization problems which have the following structure:

$$\begin{aligned}(\text{2SLP}) : \min\ & c^T x + \sum_{\omega \in \Omega} p_\omega \theta_\omega \\ \text{s.t.} & Ax \leq b \\ & Q(x,\omega) \leq \theta_\omega, \forall \omega \in \Omega \\ & x \geq 0\end{aligned}$$

where $Q(x,\omega) = \min\{q_\omega^T y_\omega \textbf{ s.t. } W_\omega y_\omega = h_\omega - T_\omega x, y_\omega \geq 0\}$. Note that, q, h, T are random parameters of appropriate size. In (2SLP), x denotes the first stage decision vector and y_ω is the second stage decision vector associated with scenario ω.

To solve (2SLP), one can solve the equivalent extensive formulation (EF) for (2SLP), which is given as follows:

$$(EF) : \min\ c^T x + \sum_{\omega \in \Omega} p_\omega q_\omega^T y_\omega$$

$$\begin{aligned}
&\text{s.t.} Ax \leq b \\
&\quad T_\omega x + W_\omega y_\omega = h_\omega, \forall \omega \in \Omega \\
&\quad x \geq 0 \ , \ y_\omega \geq 0 \ , \ \forall \omega \in \Omega
\end{aligned}$$

However, solving (EF) directly is computationally challenging even with state-of-the-art solvers available. To overcome this challenge, multi-cut L-shaped Algorithm takes advantage of the structure of (2SLP). Using the notation in (EF), we define some associated structures for the multi-cut L-shaped Algorithm as follows:

$$\begin{aligned}
(MP_{LP}) : \min \ & c^T x + \sum_{\omega \in \Omega} p_\omega \theta_\omega \\
&\text{s.t.} Ax \leq b \\
&\quad D_m^T x \geq d_m \ , \ m = 1, ..., p \\
&\quad G_{n\omega}^T x + \theta_\omega \geq g_{n\omega} \ , \omega \in \Omega, n = 1, ..., t_\omega \\
&\quad x \geq 0
\end{aligned}$$

$$\begin{aligned}
(SP_{LP}(\omega, x)) : \min & q_\omega^T y_\omega \\
&\text{s.t.} T_\omega x + W_\omega y_\omega = h_\omega \\
&\quad y_\omega \geq 0
\end{aligned}$$

Step 0: Initialize $l = p = n_1 = \ldots = n_{|\Omega|} = 0$

Step 1: Set $l = l + 1$ and solve Master Problem (MP_{LP}) to get solutions for x^l and $(\theta_\omega^l)_{\omega \in \Omega}$. Go to Step 2.

Step 2: Check feasibility for each subproblem $(SP_{LP}(\omega, x^l))$ with the first stage solution x^l, when an infeasible subproblem is found add a feasibility cut (set $p = p + 1$) and go back to Step 1. If there are no infeasible subproblems proceed to Step 3.

Step 3: For every scenario, solve the subproblem associated and get the dual solution π_ω^l. For every scenario, check if $(h_\omega - T_\omega x^l)^T \pi_\omega^l > \theta_\omega^l$, then add optimality cut (set $n_\omega = n_\omega + 1$). Whenever $(h_\omega - T_\omega x^l)^T \pi_\omega^l \leq \theta_\omega^l$ for every scenario, conclude the incumbent solution as optimal. If optimality is not concluded, go to Step 1.

As mentioned in description, multi-cut L-shaped algorithm uses two types of cuts for iterative construction of the feasible region for (2SLP).

Feasibility Cuts: Flow of these cuts are marked with red in Figure 4.1. In Step 2 of the algorithm, infeasibility in a subproblem leads to a feasibility cut. Whenever a subproblem is infeasible, its dual must be unbounded. Hence, we can find a ray in the dual feasible region. To find this ray, let σ be the dual solution corresponding to the optimal basis of the following problem:

$$\begin{aligned} &\min\ e^T w^+ + e^T w^- \\ &\text{s.t.} W y_\omega + w^+ - w^- = h_\omega - T_\omega x^l \\ &\quad y_\omega \geq 0, w^+ \geq 0, w^- \geq 0. \end{aligned}$$

Then, feasibility cut for infeasible subproblem reads as follows:

$$D^T x \geq d, \text{ where } D = T_\omega^T \sigma \text{ and } d = h_\omega^T \sigma.$$

Optimality Cuts: Flow of these cuts are marked with green in Figure 4.1. For a subproblem, if the current solution satisfies $(h_\omega - T_\omega x^l)^T \pi_\omega^l > \theta_\omega^l$, then the optimality cut reads as follows:

$$G^T x + \theta_\omega \geq g, \text{ where } G = T_\omega^T \pi_\omega^l \text{ and } g = h_\omega^T \pi_\omega^l.$$

In Figure 4.1, we illustrate the flow of information between problems used in L-shaped algorithm. At every iteration, we solve MP and get a solution for first stage variables. Then, this solution is sent to SPs and each SP is solved using this solution. If there are any SPs that is infeasible with the current first stage solution, a feasibility cut is sent to the MP and it is solved again. If all SPs are feasible, then we check the feasibility of θ_ω for each scenario. If θ_ω is infeasible for a scenario, then an optimality cut is sent to MP. Iterations continue until optimal solutions for first and second stages are found.

4.2 Improved Integer L-Shaped

In this section, we present the Improved Integer L-shaped which is an algorithm for solving two-stage stochastic integer programming problems with binary first

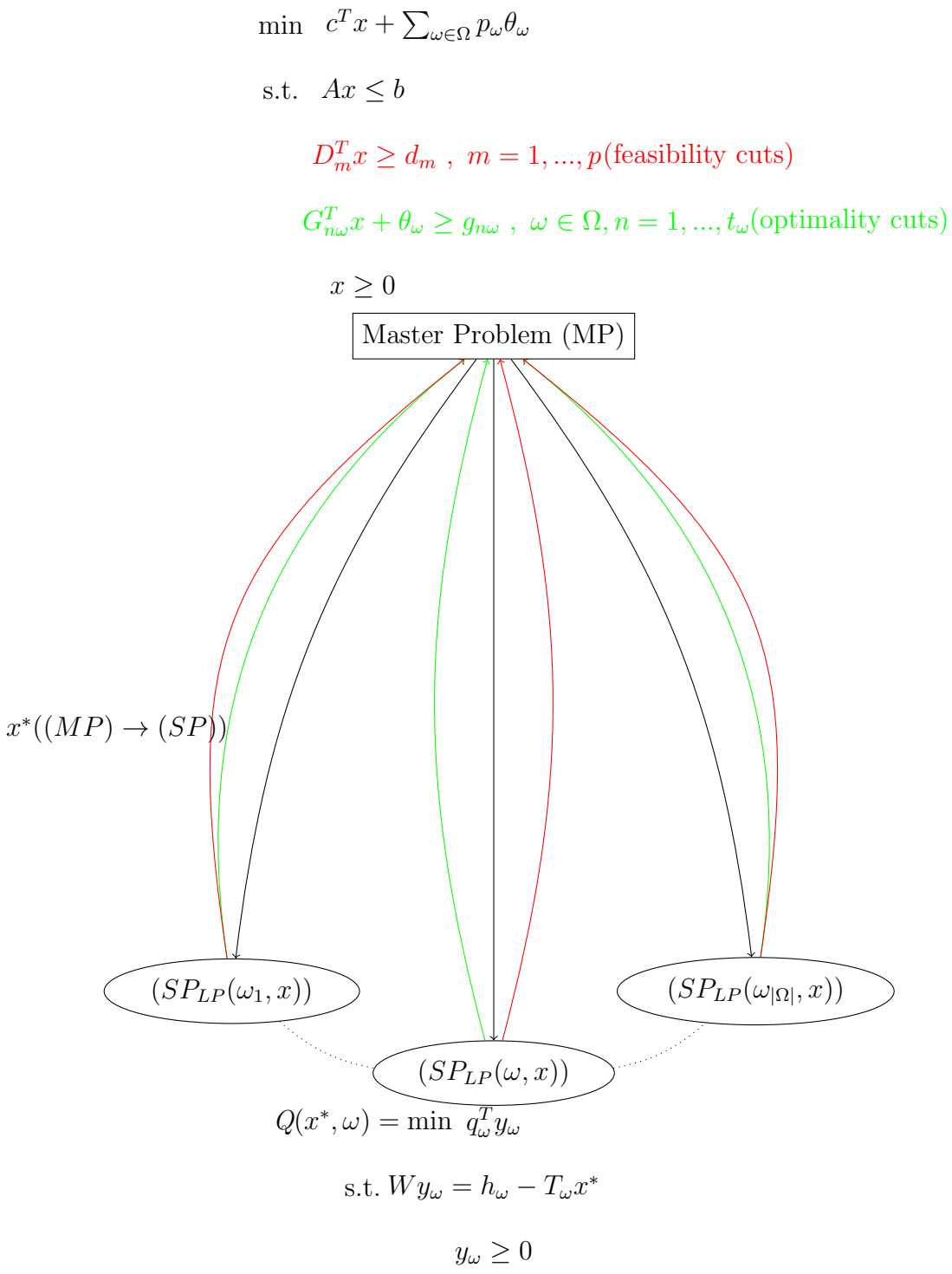


Figure 4.1: Illustration of the multi-cut L-shaped algorithm [37]. Note that the arrows represent the information flow between problems. Feasibility cuts are marked with red and optimality cuts are marked with green.

stage variables. This method can also be extended for mixed-integer first stage variables, see [7] for details. In this section, both Integer L-shaped method and Improved Integer L-shaped are described and differences are highlighted. Unlike the original description of Improved Integer L-shaped, we provide the multi-cut version of the algorithm.

To provide the algorithm statement and necessary methods for Improved Integer L-shaped, we use a similar notation for two-stage stochastic IP problems from [7]. Assuming finite probability space, suppose that the original two-stage stochastic integer programming problem is given by:

$$\begin{aligned}(\text{2SIP}) : \min\ & c^T x + \sum_{\omega \in \Omega} p_\omega \theta_\omega \\ & Ax \leq b \\ & Q(x, \omega) \leq \theta_\omega, \forall \omega \in \Omega \\ & x \in \{0, 1\}^{n_1}.\end{aligned}$$

In (2SIP), $Q(x, \omega) := \min\{q_\omega^T y_\omega : W_\omega y_\omega = h_\omega - T_\omega x, y_\omega \in \mathcal{Y}\}$. Note that, q, W, h, T are random parameters of appropriate size and $\mathcal{Y}$ is a mixed-integer domain. In (2SIP), x denotes the first stage decision vector and y_ω is the second stage decision vector. In each iteration of Integer L-shaped, $Q(x, \omega)$ is approximated by generating cuts which are known to be underestimators of $Q(x, \omega)$. Improved Integer L-shaped proposes an alternative cut generation method for this approximation procedure. Since Improved Integer L-shaped is based on the Integer L-shaped method, we first describe the standard Integer L-Shaped method in Algorithm 1. First of all, we define the objects needed to state the algorithm as follows:

Initialize the master problem (MP):

$$\begin{aligned}(\text{MP}) : \min\ & c^T x + \sum_{\omega \in \Omega} p_\omega \theta_\omega \\ & Ax \leq b \\ & \theta_\omega \geq L, \forall \omega \in \Omega \\ & x \in \{0, 1\}^{n_1}\end{aligned}$$

In the initial (MP), we relax all the constraints of type $Q(x,\omega) \leq \theta_\omega$ and introduce an appropriate lower bound L for θ_ω. Note that, L can be calculated differently for different types of problem. An example of trivial L, can be given as zero for subproblems with positive cost coefficients. Observe that, this initial (MP) corresponds to considering only first stage decision variables of the problem. In the further steps of the algorithm, relaxed constraints are partially reconstructed iteratively.

To use in intermediate steps of the algorithm, we define $Q_{LP}(x,\omega)$ as the linear programming relaxation of $Q(x,\omega)$ defined above. For a given first stage solution x^* and a scenario ω, we use the dual solution of $Q_{LP}(x^*,\omega)$, π_ω, to generate subgradient cuts of the following form:

$$G^T x + \theta_\omega \geq g, \text{ where } G = T_\omega^T \pi_\omega \text{ and } g = h_\omega^T \pi_\omega. \tag{4.1}$$

Given $Q(x^*,\omega)$, both Improved Integer L-shaped and standard Integer L-shaped method have integer optimality cuts in the following form:

$$\theta_\omega \geq (Q(x^*,\omega) - L)\Big(\sum_{i \in S(x^*)} x_i - \sum_{i \notin S(x^*)} x_i - |S(x^*)|\Big) + Q(x^*,\omega) \tag{4.2}$$

Note that, $S(x)$ is the support of the vector x i.e. elements which are non-zero are in $S(x)$. Observe that, we provide the multi-cut version of integer optimality cuts. Proof of validity for this class of cuts is the same as single-cut version, which is provided in [6]. Also note that, we set $Q(x^*,\omega)$ to a sufficiently large number when a subproblem is infeasible. Since constraints of type (4.2) only cut-off the evaluated solution, computation times of standard Integer L-shaped can be very high. Integer L-shaped method is described as follows:

Step 0: Initialize $V = \emptyset$ and $l = 0$.
Step 1: Solve LP relaxation of the problem with multi-cut L-shaped method. Keep feasibility and optimality cuts.
Step 2: Make first stage variables binary in Master Problem (MP).
Step 3: Set $l = l+1$. Solve Master Problem and get solutions for x^l and $(\theta_\omega^l)_{\omega\in\Omega}$.
Step 4: If first stage solution x^l is visited before ($x \in V$), then declare the current solution as optimal and terminate. If x^l is not visited before, solve all linear

subproblems, $Q_{LP}(x^l, \omega)$, and add associated subgradient cuts (4.1) to (MP). Proceed to Step 5.

Step 5: Set $V = V \cup \{x^l\}$. Solve all integer subproblems, $Q(x^l, \omega)$, if any of them is infeasible then cut-off the current solution by adding (4.2) to (MP). If all integer subproblems are feasible and if $\theta^l_\omega < Q(x^l, \omega)$, add integer optimality cut (4.2) to (MP). If there are any cuts added in this step, go back to Step 3. If no cuts are added in this step, then declare x^l as optimal and terminate.

In Algorithm 1, we provide a pseudo-code for Integer L-shaped method. In the pseudo-code, we use Standard Cut Generation function (Algorithm 2) which includes steps 4 and 5. Note that, this function describes how optimality cuts are added to (MP) and how optimality is concluded. Improved Integer L-shaped modifies steps 4 and 5 to enhance this algorithm by providing an alternative for this procedure.

Algorithm 1: Integer L-Shaped Method

Input : A, b, Q, Q_{LP}
Output: optimal solution $\left(x^*, (\theta_\omega)_{\omega\in\Omega}\right)$ to (2SIP) and optimal value

1 Compute a lower bound L of $Q(x, \omega)$
2 Solve LP relaxation of (2SIP) with multi-cut L-shaped Method, see Section 4.1 for details. Keep resulting feasibility and optimality cuts in (MP)
3 Declare first stage variables as binary in (MP)
4 **while** *Solved=false* **do**
5 Set $l = l + 1$
6 Solve (MP) and get solutions $\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right)$
7 Set *Solved* = Standard Cut Generation$\left(\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right), Q, Q_{LP}, V\right)$, see Algorithm 2 for details.
8 **end**
9 **return** x^* and optimal value

In Integer L-shaped method, V is defined as the set of visited nodes, which means that if $x \in V$ then $Q(x, \omega)$ is already known for every scenario and necessary optimality cuts for x are already generated before.

As mentioned earlier, Algorithms 1 and 2 describe the necessary procedures for standard Integer L-shaped method. Algorithm 4 describes the alternative cut

Algorithm 2: Standard Cut Generation

Input : $\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right), Q, Q_{LP}, V$
Output: true: if x^l is feasible, false: otherwise

```
if x^l ∈ V then
    return true
end
for ω ∈ Ω do
    Compute Q_LP(x^l, ω), i.e. solve linear subproblems
    Add subgradient cut (4.1) to (MP)
end
V = V ∪ {x^l}
for ω ∈ Ω do
    Compute Q(x^l, ω), i.e. solve integer subproblems
    if θ^l_ω < Q(x^l, ω) then
        Add integer optimality cut (4.2) to (MP)
        Set cut_added = true
    end
end
if cut_added=true then
    return false
else
    return true
end
```

generation strategy proposed by Improved Integer L-shaped. For describing the alternative strategy, we define V_{LP} as the set of first stage solutions for which the Q_{LP} is already calculated. The important idea of alternating cut strategies is to use Q_{LP} values to check the feasibility of a candidate first stage solution. This alternative eliminates the necessity to compute Q for feasibility check at each iteration. Improved Integer L-shaped can be explicitly described as follows:
Step 0: Initialize $V = V_{LP} = \emptyset$ and $l = 0$.
Step 1: Solve LP relaxation of the problem with multi-cut L-shaped method. Keep feasibility and optimality cuts.
Step 2: Make first stage variables binary in Master Problem (MP).
Step 3: Set $l = l+1$. Solve Master Problem and get solutions for x^l and $(\theta^l_\omega)_{\omega\in\Omega}$.
Step 4: If integer recourse value for first stage solution x^l is calculated before ($x^l \in V$), then declare the current solution as optimal and terminate. If this is not the case, check if linear recourse value for first stage solution x^l is calculated before ($x^l \in V_{LP}$). If it is calculated, then go to Step 5. If it is not calculated, set $V_{LP} = V_{LP} \cup \{x^l\}$ and solve all linear subproblems, $Q_{LP}(x^l, \omega)$, and if $\theta^l_\omega < Q_{LP}(x^l, \omega)$ then add associated subgradient cut (4.1) to (MP). If there are any subgradient cuts are added, go back to Step 3.
Step 5: Set $V = V \cup \{x^l\}$. Solve all integer subproblems, $Q(x^l, \omega)$, if any of them is infeasible then cut-off the current solution by adding (4.2) to (MP). If all integer subproblems are feasible and if $\theta^l_\omega < Q(x^l, \omega)$, then add integer optimality cut (4.2) to (MP). If any cuts added in this step, go back to Step 3. If no cuts are added in this step, then declare x^l as optimal and terminate.

Observe that, Improved Integer L-shaped utilizes the fact that linear recourse value is always less than or equal to integer recourse value and eliminates the need for solving integer subproblems at every iteration.

In Algorithm 3, we provide a pseudo-code for Improved Integer L-shaped. In the pseudo-code, we use Alternating Cut Generation function (Algorithm 4) which includes steps 4 and 5. Observe that, this modification provided by Algorithm 4, distinguishes Improved Integer L-shaped from standard Integer L-shaped method.

Algorithm 3: Improved Integer L-Shaped Method

Input : A, b, Q, Q_{LP}

Output: optimal solution $\left(x^*, (\theta_\omega)_{\omega\in\Omega}\right)$ to (2SIP) and optimal value

Compute a lower bound L of $Q(x, \omega)$

Solve LP relaxation of (2SIP) with multi-cut L-shaped Method, see Section 4.1 for details. Keep resulting feasibility and optimality cuts in (MP)

Declare first stage variables as binary in (MP)

while *Solved=false* **do**

  Set $l = l + 1$

  Solve (MP) and get solutions $\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right)$

  Set $Solved =$ Alternating Cut Generation$\left(\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right), Q, Q_{LP}, V, V_{LP}\right)$, see Algorithm 4 for details.

end

return x^* and optimal value

Algorithm 4: Alternating Cut Generation

Input : $\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right), Q, Q_{LP}, V, V_{LP}$

Output: true: if x^l is feasible, false: otherwise

```
if x^l ∈ V then
    return true
end
if x^l ∉ V_LP then
    V_LP = V_LP ∪ {x^l}
    for ω ∈ Ω do
        Compute Q_LP(x^l, ω), i.e. solve linear subproblems
        if θ^l_ω < Q_LP(x^l, ω) then
            Add subgradient cut (4.1) to (MP)
            lp_cut_added = true
        end
    end
    if lp_cut_added = true then
        return false
    end
end
V = V ∪ {x^l}
for ω ∈ Ω do
    Compute Q(x^l, ω), i.e. solve integer subproblems
    if θ^l_ω < Q(x^l, ω) then
        Add integer optimality cut (4.2) to (MP)
        Set cut_added = true
    end
end
if cut_added=true then
    return false
else
    return true
end
```

4.3 Decomposition of 2SATFM

In this section, we construct the objects needed for solving (2SATFM) using Improved Integer L-shaped. We explicitly define subproblems and initial Master Problem for (2SATFM). While constructing these objects, we use the same notation as Chapter 3. We use first stage time periods and associated variables to construct the master problem. As for subproblems, we define an associated subproblem for any scenario representing the recourse function at that specific scenario.

Explicit Description of Subproblems: If first stage decision vector x is given by $\{x_r^f(t) : f \in F, r \in R^f, t \in T_1 \cap T^f\}$, then $Q(x,\omega)$ is defined as:

$$\min \sum_{f\in F}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2} e_r^{ft} x_{r\omega}^f(t) \tag{4.3}$$

$$\text{s.t.} \sum_{r\in R^f}\sum_{t\in T^f\cap T_1} x_r^f(t) + \sum_{r\in R^f}\sum_{t\in T^f\cap T_2} x_{r\omega}^f(t) = 1, \quad \forall f \in F \tag{4.4}$$

$$\sum_{f\in F:a^f=k}\sum_{r\in R^f}\Big(x_r^f(t-t_{rk}^f)1_{\{t-t_{rk}^f\in T_1\}} + x_{r\omega}^f(t-t_{rk}^f)1_{\{t-t_{rk}^f\in T_2\}}\Big) \le A_{k\omega}^t,$$

$$\forall k \in A, t \in T_2 \tag{4.5}$$

$$\sum_{f\in F:d^f=k}\sum_{r\in R^f} x_{r\omega}^f(t) \le D_{k\omega}^t, \quad \forall k \in A, t \in T_2 \tag{4.6}$$

$$\sum_{f\in F}\sum_{r\in R^f:s\in r}\Big(x_r^f(t-t_{rs}^f)1_{\{t-t_{rs}^f\in T_1\}} + x_{r\omega}^f(t-t_{rs}^f)1_{\{t-t_{rs}^f\in T_2\}}\Big) \le C_{s\omega}^t,$$

$$\forall s \in S, t \in T_2 \tag{4.7}$$

$$\sum_{r\in R^f}\sum_{\tau\ge t-t_{ra^f}^f-\delta}\Big(x_r^{f'}(\tau)1_{\tau\in T_1} + x_{r\omega}^{f'}(\tau)1_{\tau\in T_2}\Big)$$

$$+\sum_{r\in R^{f'}}\sum_{\tau\le t}\Big(x_r^{f'}(\tau)1_{\tau\in T_1} + x_{r\omega}^{f'}(\tau)1_{\tau\in T_2}\Big) \le 1, \forall (f,f') \in \mathcal{C}, t \in T \tag{4.8}$$

$$\sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} x_r^f(t)\big[t+t_{ra^f}^f-\tau^f\big]_+$$

$$+\sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{t\in T^f\cap T_2} x_{r\omega}^f(t)\big[t+t_{ra^f}^f-\tau^f\big]_+ \le \mathcal{M}, \forall k \in \mathcal{A} \tag{4.9}$$

$$x_{r\omega}^f(t) \in \{0,1\}, \quad \forall f \in F, r \in R^f, t \in T^f \cap T_2 \tag{4.10}$$

To obtain $Q_{LP}(x,\omega)$, use the linear relaxations of constraints defined by (4.10), i.e. $x^f_{r\omega}(t) \in [0,1]$.

Constructing the Master Problem: For first stage decision variables $\{x^f_r(t) : f \in F, r \in R^f, t \in T_1 \cap T^f\}$, the Master Problem for (2SATFM) is given by:

$$\min \sum_{f\in F}\sum_{r\in R^f}\sum_{t\in T^f\cap T_1} e^{ft}_r x^f_r(t) + \sum_{\omega\in\Omega} p_\omega \theta_\omega \tag{4.11}$$

$$\text{s.t.} \sum_{f\in F:a^f=k}\sum_{r\in R^f} x^f_r(t - t^f_{rk}) \le A^t_k, \quad \forall k \in A, t \in T_1 \tag{4.12}$$

$$\sum_{f\in F:d^f=k}\sum_{r\in R^f} x^f_r(t) \le D^t_k, \quad \forall k \in A, t \in T_1 \tag{4.13}$$

$$\sum_{f\in F}\sum_{r\in R^f:s\in r} x^f_r(t - t^f_{rs}) \le C^t_s, \quad \forall s \in S, t \in T_1 \tag{4.14}$$

$$x^f_r(t) \in \{0,1\}, \quad \forall f \in F, r \in R^f, t \in T^f \cap T_1 \tag{4.15}$$

Observe that, flight connectivity (3.14) and maximum delay constraints (3.19) are not present in the master problem explicitly. Since (3.14) and (3.19) cannot be written without the second stage variables, they are removed while constructing the master problem. Note that, even though (3.14) and (3.19) are removed from master problem, we consider these constraints in their respective subproblems.

4.4 Partial Benders' Approach

This section provides a description of Partial Benders' approach proposed by Crainic et al. [8]. We define scenario selection strategies and their correspondence to (2SATFM) problem. Some necessary modifications for the master problem of (2SATFM) are specified and explicitly described.

Scenario selection strategy in Partial Benders' approach, proposes to include a subset of subproblems to master problem. By including a subset of scenarios to master problem, this method eliminates a significant portion of feasibility and optimality cuts. However, including subproblems to master problem enlarges the

master problem and each iteration becomes computationally more demanding. Hence, effective selection of subproblems are essential for advantageous utilization of this approach. To describe the algorithm and scenario selection strategies, we define the following:

For scenario selection: We define the set of selected scenarios as $\tilde{\Omega} \subset \Omega$. This set represents the scenarios to be included in the Master Problem. To select scenarios and construct the set $\tilde{\Omega}$, we define the "covering" relation using associated subproblems. For two scenarios ω and $\tilde{\omega}$, we say that ω covers $\tilde{\omega}$ if the feasibility of the subproblem associated with ω implies the feasibility of the subproblem associated with $\tilde{\omega}$.

Modified version of the Master Problem for using Partial Benders': Given a subset of scenarios, $\tilde{\Omega}$, to include in the Master Problem, we define the initial master problem for Partial Benders' as follows:

$$
\begin{aligned}
(MP_{partial}) : \min\ & c^T x + \sum_{\omega \in \Omega} p_\omega \theta_\omega \\
& Ax \leq b \\
& \theta_\omega = q_\omega^T y_\omega\ ,\ \forall \omega \in \tilde{\Omega} \\
& W_\omega y_\omega = h_\omega - T_\omega x, \forall \omega \in \tilde{\Omega} \\
& y_\omega \in \mathcal{Y}, \forall \omega \in \tilde{\Omega} \\
& x \in \{0,1\}^{n_1}.
\end{aligned}
$$

Observe that, the resulting first stage solution must be feasible for every scenario in $\tilde{\Omega}$. This observation and above definition of "covering" relation, leads to the main idea of the Partial Benders' approach. Since this approach defines covering across scenarios using feasibility implications, the number of feasibility cuts needed is expected to decrease drastically.

After defining the required concepts, we elaborate on a strategy for selecting scenarios for $\tilde{\Omega}$. For selecting the scenarios to include in the set $\tilde{\Omega}$, we use the covering formulation technique proposed in [8]. The main idea of this technique is to "cover" a larger number of scenarios with smallest possible $\tilde{\Omega}$. To achieve this, we formulate a simple optimization problem. We define the decision variables

and parameters for this problem as follows: **Sets**:

$\mathcal{C}_\omega$: Set of scenarios "covering" the scenario ω

Parameters:

γ : Cost associated to including a scenario to master problem

Decision Variables:

$$y_\omega = \begin{cases} 1 & \text{if } \omega \text{ is covered} \\ 0 & \text{otherwise} \end{cases}$$

$$x_\omega = \begin{cases} 1 & \text{if } \omega \text{ is included in MP} \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned} \text{(SSP)} \max & \sum_{\omega \in \Omega} y_\omega - \gamma \sum_{\omega \in \Omega} x_\omega \\ \text{s.t.} & \sum_{\omega' \in \mathcal{C}_\omega} x_{\omega'} \geq y_\omega, \forall \omega \in \Omega, \\ & x_\omega, y_\omega \text{ binary }, \forall \omega \in \Omega \end{aligned}$$

Scenario selection problem (SSP) constructs the smallest possible $\tilde{\Omega}$ by covering largest possible number of scenarios in Ω. Note that, γ is positive.

In the case of (2SATFM) problem, there are fixed recourse and technology, i.e. T and W matrices do not vary across scenarios. Hence, any subproblem can be distinctly defined with h_ω as follows:

$$Q(x, \omega) = \min\{q^T y_\omega \textbf{ s.t. } W y_\omega = h_\omega - Tx, y_\omega \in \mathcal{Y}\}.$$

Hence, definition of "covering" reduces to ω covers ω' if $h_\omega \leq h_{\omega'}$. This implies that $\mathcal{C}_\omega$ requires trivial effort to calculate for (2SATFM) problem.

Combining Partial Benders' approach with Improved Integer L-shaped, only modifies the set of subproblems used in cut generation procedures. Which implies that, each iteration requires the computation of less number of subproblems. This modification also results in a less number of cuts. So that, growth of Master Problem size is slower compared to both Improved Integer L-shaped and standard Integer L-shaped method.

The modified algorithm can be explicitly described as follows:

Step 0: Initialize $V = V_{LP} = \emptyset$ and $l = 0$.
Step 1: Solve LP relaxation of the problem with L-shaped method by using the master problem $MP_{partial}$. Keep feasibility and optimality cuts.
Step 2: Make necessary variables binary in Master Problem ($MP_{partial}$).
Step 3: Set $l = l + 1$. Solve ($MP_{partial}$) and get solutions for x^l and $(\theta^l_\omega)_{\omega\in\Omega}$.
Step 4: If integer recourse value for first stage solution x^l is calculated before ($x^l \in V$), then declare the current solution as optimal and terminate. If this is not the case, check if linear recourse value for first stage solution x^l is calculated before ($x^l \in V_{LP}$). If it is calculated, then go to Step 5. If it is not calculated, set $V_{LP} = V_{LP} \cup \{x^l\}$ and solve linear subproblems associated to scenarios in the set $\Omega\backslash\tilde{\Omega}$, and if $\theta^l_\omega < Q_{LP}(x^l, \omega)$ then add associated subgradient cut (4.1) to ($MP_{partial}$). If there are any subgradient cuts added, go back to Step 3.
Step 5: Set $V = V \cup \{x^l\}$. Solve integer subproblems associated to scenarios in the set $\Omega\backslash\tilde{\Omega}$, if any of them is infeasible then cut-off the current solution by adding (4.2) to ($MP_{partial}$). If considered integer subproblems are feasible and if $\theta^l_\omega < Q(x^l, \omega)$, then add integer optimality cut (4.2) to ($MP_{partial}$). If there are any cuts added in this step, go back to Step 3. If no cuts are added in this step, then declare x^l as optimal and terminate.

Note that, the combination of Partial Benders' approach with Improved Integer L-shaped, includes changes in multi-cut L-shaped method in Step 1 as well. In Algorithm 5, we provide a pseudo-code for the modified Improved Integer L-shaped using the cut generation function described in Algorithm 6. Observe that, Algorithms 5 and 6 are modified versions of Algorithms 3 and 4, using Partial Benders' approach.

Algorithm 5: Improved Integer L-Shaped Method with Partial Benders'

Input : A, b, Q, Q_{LP}

Output: optimal solution $\left(x^*, (\theta_\omega)_{\omega\in\Omega}\right)$ to (2SIP) and optimal value

Compute a lower bound L of $Q(x, \omega)$

Solve LP relaxation of (2SIP) with multi-cut L-shaped Method by using $(MP_{partial})$ as the master problem, see Section 4.1 for details. Keep resulting feasibility and optimality cuts in $(MP_{partial})$

Declare first stage variables as binary in $(MP_{partial})$

while *Solved=false* **do**

  Set $l = l + 1$

  Solve $(MP_{partial})$ and get solutions $\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right)$

  Set *Solved* =

  Alternating Cut Generation 2$\left(\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right), Q, Q_{LP}, V, V_{LP}\right)$, see Algorithm 4 for details.

end

return x^* and optimal value

Algorithm 6: Alternating Cut Generation 2 (with Partial Benders')

Input : $\left(x^l, (\theta^l_\omega)_{\omega\in\Omega}\right), Q, Q_{LP}, V, V_{LP}$

Output: true: if x^l is feasible, false: otherwise

```
if x^l ∈ V then
    return true
end
if x^l ∉ V_LP then
    V_LP = V_LP ∪ {x^l}
    for ω ∈ Ω\Ω̃ do
        Compute Q_LP(x^l, ω), i.e. solve linear subproblems
        if θ^l_ω < Q_LP(x^l, ω) then
            Add subgradient cut (4.1) to (MP)
            lp_cut_added = true
        end
    end
    if lp_cut_added = true then
        return false
    end
end
V = V ∪ {x^l}
for ω ∈ Ω\Ω̃ do
    Compute Q(x^l, ω), i.e. solve integer subproblems
    if θ^l_ω < Q(x^l, ω) then
        Add integer optimality cut (4.2) to (MP)
        Set cut_added = true
    end
end
if cut_added=true then
    return false
else
    return true
end
```

4.5 Approximation for CVaR-constrained Problem

When we introduce CVaR constraints in linear form to (2SATFM) problem, decomposable structure of (2SATFM) is lost due to the constraint given by (3.23). Since (3.23) contains decision variables from every scenario, we cannot use Improved Integer L-shaped and Partial Benders' methods to obtain an exact solution to the CVaR-constrained problem. In this section, we propose an approximation scheme for CVaR-constrained problem using Partial Benders' approach and specific structures observed in (2SATFM).

Recall that, we define $\tilde{\Omega}$ to be the set of scenarios included in the master problem for Partial Benders' described in Section 4.4. We assume that for any scenario $\omega \in \Omega$, there exists $\tilde{\omega} \in \tilde{\Omega}$ such that $\tilde{\omega}$ "covers" ω. Which means that, for any scenario $\omega \in \Omega$, there exists a scenario $\tilde{\omega} \in \tilde{\Omega}$ such that $h_{\tilde{\omega}} \leq h_{\omega}$.

We adjust the probabilities of each $\tilde{\omega} \in \tilde{\Omega}$ tracing to $\tilde{\Omega}$ as follows:

$$\bar{p}_{\tilde{\omega}} := \frac{p_{\tilde{\omega}}}{\sum_{\omega \in \tilde{\Omega}} p_{\omega}}$$

Then we define a random variable on $\tilde{\Omega}$ representing total delay for an airline $k \in \mathcal{A}$ as follows:

$$X_k(\omega) := \sum_{f \in \bar{\mathcal{A}}_k} \sum_{r \in R^f} \sum_{t \in T^f \cap T_1} x_r^f(t) \left[t + t_{ra^f}^f - \tau^f\right]_+ + \sum_{f \in \bar{\mathcal{A}}_k} \sum_{r \in R^f} \sum_{t \in T^f \cap T_2} x_{r\omega}^f(t) \left[t + t_{ra^f}^f - \tau^f\right]_+$$

Theorem 4.5.1. *For $\omega \in \Omega \backslash \tilde{\Omega}$ define,*

$$\Psi(\omega) = \arg\min_{\tilde{\omega} \in \tilde{\Omega}} \Big\{ X_k(\tilde{\omega}) : X_k(\tilde{\omega}) \geq Y_k(\omega) \Big\}$$

$$\bar{P} = \sum_{\omega \in \tilde{\Omega}} p_\omega$$

Assume that, $\Psi(\omega)$ exists for each $\omega \in \Omega \backslash \tilde{\Omega}$ and the following holds for every $\omega \in \tilde{\Omega}$:

$$\frac{p_\omega}{\bar{P}} \geq \sum_{\tilde{\omega} \in \Omega \backslash \tilde{\Omega} : \Psi(\tilde{\omega}) = \omega} \frac{p_{\tilde{\omega}}}{1 - \bar{P}}$$

Then,

$$CVaR_\alpha(X_k) \geq CVaR_\alpha(Y_k)$$

Proof. We analyze systems of $\eta, (w_\omega)_{\omega \in \Omega}, x$ corresponding to $CVaR_\alpha(X_k) \leq \beta$ and $CVaR_\alpha(Y_k) \leq \beta$. Then, derive the conditions for $CVaR_\alpha(X_k) \geq CVaR_\alpha(Y_k)$ relation to hold. We first assume that there exists $\eta, (w_\omega)_{\omega \in \tilde{\Omega}}, x$ satisfying the following system:

$$\eta + \frac{1}{1-\alpha} \sum_{\omega \in \tilde{\Omega}} \bar{p}_\omega w_\omega \leq \beta$$

$$\sum_{f \in \bar{\mathcal{A}}_k} \sum_{r \in R^f} \sum_{t \in T^f} x_r^f(t) \big[t + t_{ra^f}^f - \tau^f \big]_+ - \eta \leq w_\omega, \forall \omega \in \tilde{\Omega}$$

$$0 \leq w_\omega, \forall \omega \in \tilde{\Omega}$$

$$\eta \in \mathbb{R}$$

For $\omega \in \Omega \backslash \tilde{\Omega}$, we define the $(w_\omega)_{\omega \in \Omega \backslash \tilde{\Omega}}$ as follows:

$$w_\omega := (Y_k(\omega) - \eta)_+$$

For $\omega \in \Omega \backslash \tilde{\Omega}$, defined $(w_\omega)_{\omega \in \Omega \backslash \tilde{\Omega}}$ trivially satisfies the following:

$$\sum_{f \in \bar{\mathcal{A}}_k} \sum_{r \in R^f} \sum_{t \in T^f} x_r^f(t) \big[t + t_{ra^f}^f - \tau^f \big]_+ - \eta \leq w_\omega, \forall \omega \in \Omega \backslash \tilde{\Omega}$$

$$0 \leq w_\omega, \forall \omega \in \Omega \backslash \tilde{\Omega}$$

Since we know that,

$$\eta + \frac{1}{1-\alpha} \sum_{\omega \in \tilde{\Omega}} \bar{p}_\omega w_\omega \leq \beta$$

$$\bar{P}\eta + \frac{1}{1-\alpha}\sum_{\omega\in\tilde{\Omega}} p_\omega w_\omega \leq \bar{P}\beta$$

$$\text{For } \eta + \frac{1}{1-\alpha}\sum_{\omega\in\Omega} p_\omega w_\omega \leq \beta \text{ to hold,}$$

$$(1-\bar{P})\eta + \frac{1}{1-\alpha}\sum_{\omega\in\Omega\backslash\tilde{\Omega}} p_\omega w_\omega \leq (1-\bar{P})\beta$$

$$\frac{1}{1-\alpha}\sum_{\omega\in\Omega\backslash\tilde{\Omega}} \frac{p_\omega}{1-\bar{P}} w_\omega \leq \beta - \eta$$

We want the following to hold:

$$\frac{1}{1-\alpha}\sum_{\omega\in\tilde{\Omega}} \frac{p_\omega}{\bar{P}} w_\omega \leq \beta - \eta \implies \frac{1}{1-\alpha}\sum_{\omega\in\Omega\backslash\tilde{\Omega}} \frac{p_\omega}{1-\bar{P}} w_\omega \leq \beta - \eta$$

$$\frac{1}{1-\alpha}\sum_{\omega\in\tilde{\Omega}} \frac{p_\omega}{\bar{P}} w_\omega \geq \frac{1}{1-\alpha}\sum_{\omega\in\Omega\backslash\tilde{\Omega}} \frac{p_\omega}{1-\bar{P}} w_\omega$$

For any $\tilde{\omega} \in \Omega\backslash\tilde{\Omega}$, by definition of $\Psi(\tilde{\omega})$, the following inequality is satisfied $w_{\Psi(\tilde{\omega})} \geq w_{\tilde{\omega}}$.

So for any non-included scenario, we have an included scenario in $\tilde{\Omega}$. Then for any scenario $\omega \in \tilde{\Omega}$ included, we have the set of scenarios

$$\mathcal{U}(\omega) := \{\tilde{\omega} \in \Omega\backslash\tilde{\Omega} : \Psi(\tilde{\omega}) = \omega\}$$

$$\sum_{\omega\in\tilde{\Omega}} \Big(\sum_{\tilde{\omega}\in\mathcal{U}(\omega)} \frac{p_{\tilde{\omega}}}{1-\bar{P}} \Big) w_\omega \geq \sum_{\omega\in\tilde{\Omega}} \Big(\sum_{\tilde{\omega}\in\mathcal{U}(\omega)} \frac{p_{\tilde{\omega}}}{1-\bar{P}} w_{\tilde{\omega}} \Big)$$

For the desired condition to hold, we need the following:

$$\sum_{\omega\in\tilde{\Omega}} \frac{p_\omega}{\bar{P}} w_\omega \geq \sum_{\omega\in\tilde{\Omega}} \Big(\sum_{\tilde{\omega}\in\Omega\backslash\tilde{\Omega}:\Psi(\tilde{\omega})=\omega} \frac{p_{\tilde{\omega}}}{1-\bar{P}} \Big) w_\omega$$

By the condition of this theorem, the following is true.

$$\sum_{\omega\in\tilde{\Omega}} \frac{p_\omega}{\bar{P}} \geq \sum_{\omega\in\tilde{\Omega}} \Big(\sum_{\tilde{\omega}\in\Omega\backslash\tilde{\Omega}:\Psi(\tilde{\omega})=\omega} \frac{p_{\tilde{\omega}}}{1-\bar{P}} \Big)$$

$$\sum_{\omega\in\tilde{\Omega}} \frac{p_\omega}{\bar{P}} w_\omega \geq \sum_{\omega\in\tilde{\Omega}} \Big(\sum_{\tilde{\omega}\in\Omega\backslash\tilde{\Omega}:\Psi(\tilde{\omega})=\omega} \frac{p_{\tilde{\omega}}}{1-\bar{P}} \Big) w_\omega$$

□

Condition of Theorem 4.5.1 implies that, if scenarios associated with smaller valued delays cover a larger portion of the scenarios compared to larger valued delays, then this approximation provides a valid upper bound for (R2SATFM) problem. To get an approximate solution for CVaR-constrained problem using Theorem 4.5.1, we include the following constraints to Partial Benders' Master Problem.

$$\eta^k + \frac{1}{1-\alpha}\sum_{w\in\tilde{\Omega}} \bar{p}_\omega w^k_\omega \leq \beta \ , \ \forall k \in \mathcal{A}$$
$$w^k_\omega \geq \sum_{f\in\bar{\mathcal{A}}_k}\sum_{r\in R^f}\sum_{i\in N: i\in\Phi(\omega)} x^f_{ri}\left[t + t^f_{ra^f} - \tau^f\right]_+, \forall\omega\in\tilde{\Omega}, \forall k\in\mathcal{A}$$
$$w^k_\omega \geq 0, \forall\omega\in\tilde{\Omega}, \forall k\in\mathcal{A}$$
$$\eta^k \in \mathbb{R} \ , \ \forall k \in \mathcal{A}.$$

When these constraints are included in the Master Problem, the optimal solution obtained from the Algorithm 5 described in Section 4.4, is proved to provide a feasible approximation for (R2SATFM) problem.

Chapter 5

Realistic Data Generation

This chapter presents detailed descriptions of realistic data generation for (2SATFM) problem and illustrations of generated data. We describe the generation procedure for each parameter of the problem in an orderly manner. In particular, since we generate some of the parameters using other parameters, we describe the procedures in a specific order. In Section 5.1, we elaborate on the generation of different elements in the air traffic system. In Section 5.2, we describe the construction of weather induced uncertainty for (2SATFM) problem.

5.1 Airspace Elements, Flights and Routes

Similar to most of transportation systems, air traffic systems are intrinsically associated to an underlying graph structure. Although (2SATFM) model does not explicitly require a graph, capacities of different airspace elements and routes must be generated using a graph for accurate representations of real-life systems.

Since connections between neighboring airspace elements can be conveniently modeled using a grid structure, we generate a two-dimensional grid graph, defined by $G = (V, E)$, consisting of $|A| + |S|$ vertices, where each vertex represents an airspace element of consideration. After generating the underlying graph,

we randomly label $|A|$ vertices as airports then label the remaining vertices as sectors. Since any route between two airports should contain at least one sector, we manually remove edges between any two vertices labeled as airports. See Figure 5.1 for illustrations of randomly generated air traffic networks.

Creation of flights are performed by randomly generating origin-destination pairs in the form of $(a_1, a_2) \in A \times A : a_1 \neq a_2$. After creating the set of flights F, we construct the set R^f by finding five shortest paths available for each $f \in F$ using the underlying graph G. To construct the routes as sequences of sector-time pairs defined in Chapter 3, we assume that each arc takes one time period to travel.

5.2 Stochastic Capacities

While generating stochastic capacities for different airspace elements, correspondence with observed data is essential. According to EUROCONTROL [38], non-planned capacity reductions in European Air Traffic System are almost always caused by weather conditions. To incorporate the weather condition causality in the instances, we use the weather-induced uncertainty structure proposed by Bertsimas and Gupta [25]. Authors use step functions to model time-indexed capacity changes and propose a quantitative framework capturing geographical and time-dependent structure of stochastic capacities of airspace elements. To generate a weather-induced uncertainty structure on capacity levels of airspace elements, we describe a similar procedure.

When there are no disruptions of weather events, airspace elements operate under a constant planned capacity. This constant capacity is called nominal capacity. In particular, consider an initial capacity vector $\mathbf{C} = (\bar{C}, \ldots, \bar{C}) \in \mathbb{R}^{|T|}$ where $\bar{C}$ is the nominal capacity. To take weather into account, define W as the set of possible weather events. A weather event $i \in W$, is identified by a specific origin node $s_i \in V$, a starting time T^i_{start}, a duration T_i and a reduction factor α_i. We assume that, a weather event $i \in W$ only affects the neighboring nodes

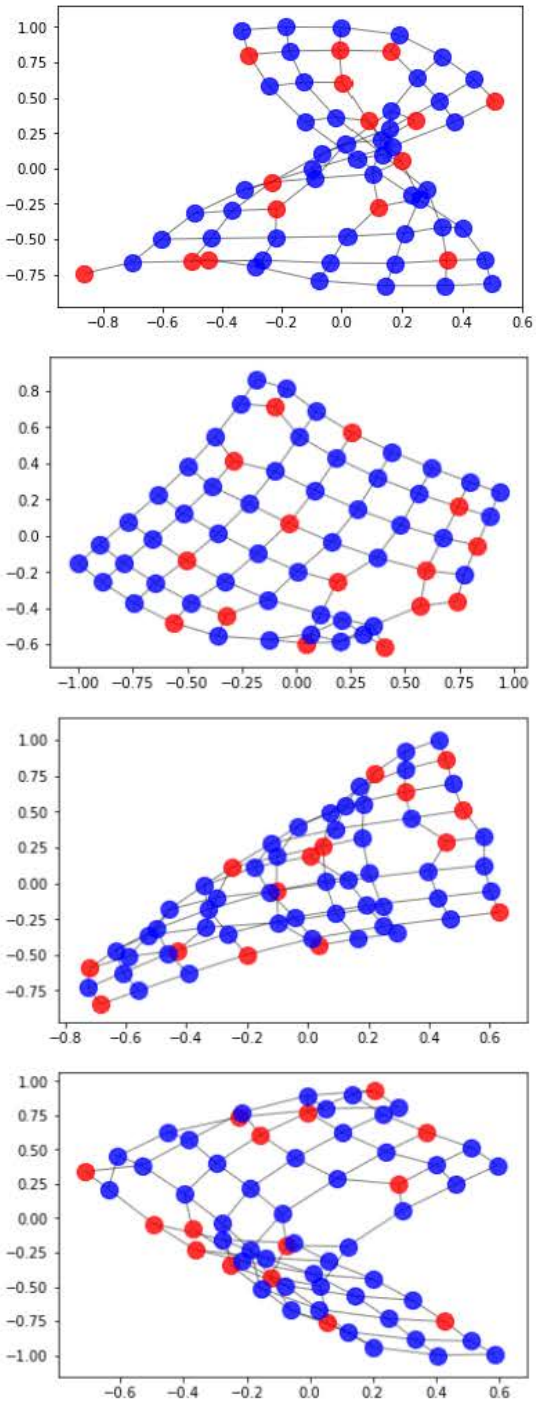


Figure 5.1: Examples of generated air traffic networks consisting of airports (red) and en-route sectors (blue) where arcs represent if there is a connection between elements

of s_i, therefore we identify the airports and sectors affected by the weather event using the underlying graph G. If $a \in A$ (or $s \in S$) is affected by the weather event $i \in W$, the capacity change of that element can be described as follows:

$$\mathbf{C}^t_{new} = \begin{cases} \bar{C} & \text{if } t < T^i_{start} \\ \alpha_i \bar{C} & \text{if } T^i_{start} \leq t \leq T^i_{start} + T_i \\ \bar{C} & T^i_{start} + T_i < t \leq |T| \end{cases}$$

After defining preliminary structures, we provide a pseudo-code for weather event generation in Algorithm 7.

Algorithm 7: Weather Event Generation

Input: G, W, A, S
Output: $(\alpha_i)_{i \in W}, (T^i_{start})_{i \in W}, (s_i)_{i \in W}, (T_i)_{i \in W}$
for $i \in W$
  Randomly select a node in V and label it as s_i
  Label neighbor nodes of s_i as affected
  Construct A_i (airports affected) and S_i (sectors affected) from neigbors
  Generate α_i from $Uniform(0, 1)$
  Generate T^i_{start} from discrete uniform over 2^{nd} stage time periods
  Generate T_i from discrete uniform over 0 to $|T| - T^i_{start}$
end for

Weather events construct capacity profiles for each airspace element using aforementioned parameters while considering geographical factors and time-dependency. See Figure 5.2 for illustrations of effects of different weather events on different capacity profiles. In this figure, we illustrate examples of $\Delta^i \mathbf{C}_e$ vectors for an arbitrary sector s and a weather event i. We graph the effect of a weather event on different sectors with respect to time.

Weather events are associated with scenarios by being randomly included in weather event sets of scenarios. To consider weathers with scenarios, we define $S_\omega \in \mathcal{P}(W)$ as the set of weather events associated with scenario ω, i.e. for each scenario we have a subset of $|W|$ different weather events. Also, for each

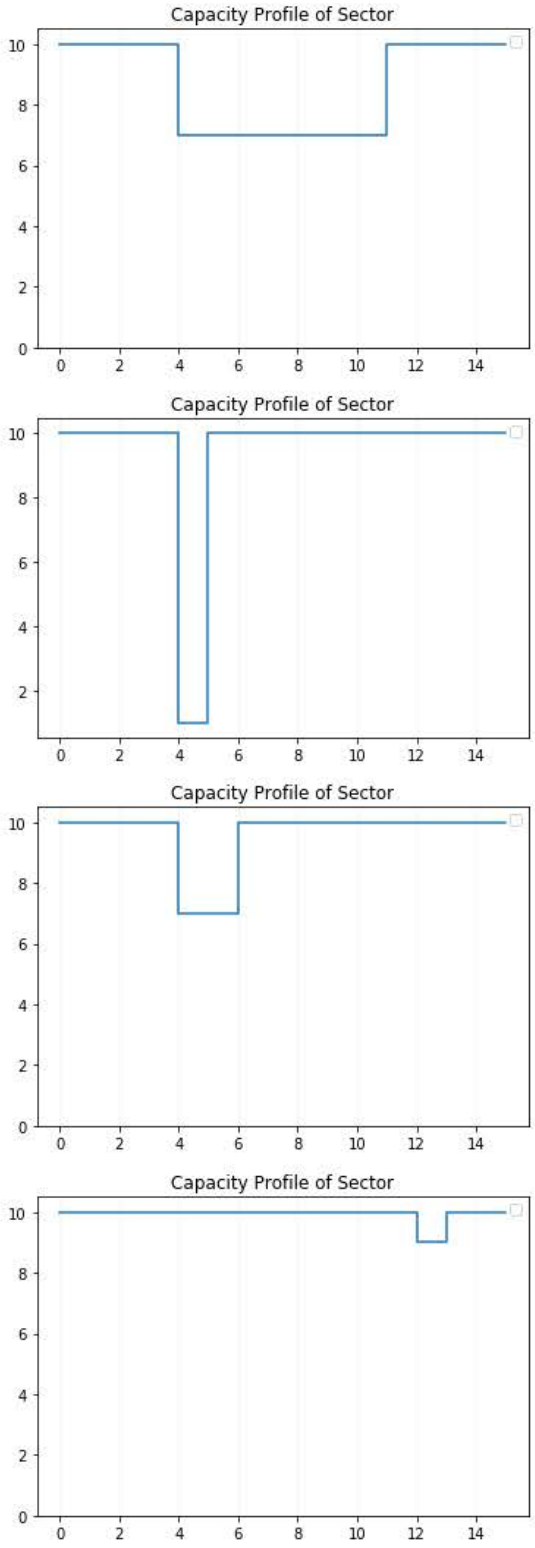


Figure 5.2: Examples of capacity profiles under different weather events

scenario, we combine the effects of the weather events in an additive manner, i.e. we sum all reductions in an airspace element's capacity profile over different weather events.

Recall that, $C^t_{s\omega}$ is the capacity of the sector s at the time t under scenario ω and $A^t_{k\omega}/D^t_{k\omega}$ represents the arrival/departure capacity of airport k at the time t under scenario ω. Let C_A/C_S be the nominal capacity value for airports/sectors. To calculate capacity profiles of different sectors under different scenarios, we first calculate capacity decrease vector $\Delta^i\mathbf{C}_e \in \mathbb{R}^{|T|}$ for each weather event $i \in W$ and for each airspace element $e \in A \cup S$ as follows.

For sector $s \in S$, time $t \in T$ and weather event $i \in W$,

$$\Delta^i\mathbf{C}^t_s = \begin{cases} 0 & \text{if } t < T^i_{start} \\ (1-\alpha_i)C_S & \text{if } T^i_{start} \le t \le T^i_{start} + T_i \text{ and } s \in S_i \\ 0 & \text{if } T^i_{start} \le t \le T^i_{start} + T_i \text{ and } s \notin S_i \\ 0 & \text{if } T^i_{start} + T_i < t \le |T| \end{cases}.$$

For airport $a \in A$, time $t \in T$ and weather event $i \in W$,

$$\Delta^i\mathbf{C}^t_a = \begin{cases} 0 & \text{if } t < T^i_{start} \\ (1-\alpha_i)C_A & \text{if } T^i_{start} \le t \le T^i_{start} + T_i \text{ and } a \in A_i \\ 0 & \text{if } T^i_{start} \le t \le T^i_{start} + T_i \text{ and } a \notin A_i \\ 0 & \text{if } T^i_{start} + T_i < t \le |T| \end{cases}.$$

Then for every airspace element $e \in A \cup S$ and every weather event $i \in W$, we have the capacity change vector $\Delta^i\mathbf{C}_e := (\Delta^i\mathbf{C}^t_e)_{t\in T} \in \mathbb{R}^{|T|}$. Calculating individual contributions of weather events to capacity profiles as vectors allows us to represent scenario-dependent capacities in a compact sense. For $\omega \in \Omega$ and $e \in A \cup S$, we represent the capacity profile $C_e(\omega) \in \mathbb{R}^{|T|}$ of airspace element e under scenario ω as follows:

$$C_e(\omega) = \bar{C} - \sum_{i \in S_\omega} \Delta^i\mathbf{C}_e, \qquad \text{where } \bar{C} = \underbrace{(C_A, \ldots, C_A)}_{|T| \text{ time periods}} \text{ or } \underbrace{(C_S, \ldots, C_S)}_{|T| \text{ time periods}}.$$

See Figure 5.3 for illustrations of different capacity profiles under different scenarios. In this figure, we illustrate arbitrary examples of $C_s(\omega)$, i.e. for a given scenario we graph the capacity with respect to time.

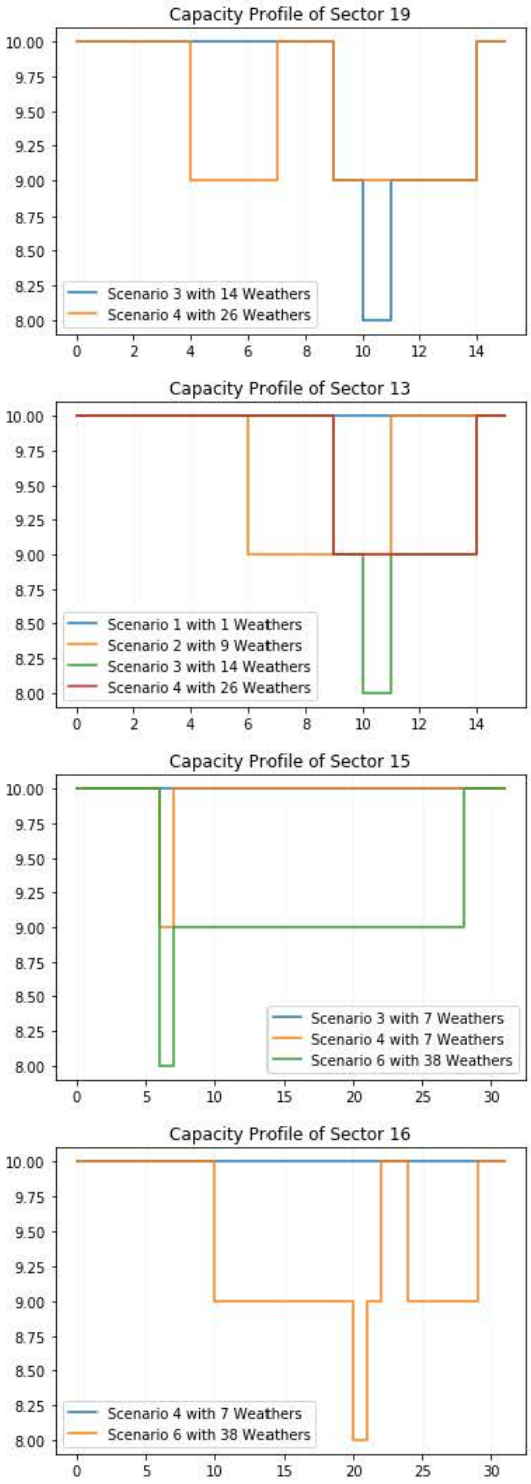


Figure 5.3: Examples of capacity profiles of different sectors under different scenarios

Chapter 6

Computational Experiments

In this chapter, we conduct computational experiments on different sizes of the problem and demonstrate the performances of algorithms discussed in Chapter 4. In all computational experiments, we use instances reflecting real-world properties of the air traffic system generated using techniques described in Chapter 5.

All computations are performed on a Windows PC with a 3.70 GHz Intel Core i5 Processor having six physical and six virtual cores and 16GB of RAM. We generate our data using Python 3.7 and use CPLEX 12.10 Python API for solving subproblems and master problems inside the algorithms. Also note that, all solution times are reported in terms of wall-clock times.

6.1 Description of Instances

Throughout this chapter, we use the following notation to identify generated instances used, to analyze computational aspects of this problem:

Notation

F : number of flights

T : number of time periods

S : number of enroute sectors

A : number of airports

r : cost of unit travel

d : cost of unit delay

VSS : Value of Stochastic Solution (as percentage)

C_A : nominal capacity for airports

C_S : nominal capacity for sectors

$|\mathcal{C}|$: number of pair of flights that are connected

$\mathcal{M}$: maximum delay allowed for total airline delay

c_r : maximum ratio of decrease in capacity for a weather event (between 0 and 1)

NW : number of weather events

$|\tilde{\Omega}|$: number of scenarios included in master problem

γ : input parameter for scenario selection

FS : number of time periods in first stage

Cov : total number of scenarios covered by scenarios in $\tilde{\Omega}$

We name instances of air traffic systems after the parameters that we use while generating them, as follows:

$$\text{Instance Name } : F \_ T \_ |\Omega| \_ S \_ A \_ FS \_ c\_r \_ NW$$

6.2 Value of Stochastic Solution (VSS)

In this section, we demonstrate the advantages of using a stochastic environment instead of a deterministic one. While analyzing the behavior of VSS for the two-stage problem, we vary relevant parameters which may affect the variation of decisions across scenarios. We use smaller instances, 100-200 flights and 16 scenarios, to present the relationship between VSS and different parameters. Generated parameters are adjusted for the scalability of smaller instances

to achieve realistic systems. For each parameter of the environment, we generate multiple problem data and analyze the general behavior.

We observe in Figure 6.1 that, as delay cost to route cost ratio increases, VSS increases. This result is expected since stochastic capacities invoke the possibility for more delays, and costlier delays are avoided when the stochastic problem is solved. Additionally, when delays become more and more costly, the value of avoiding them tends to be higher.

In some cases, even with similar flight control decisions, changes in delay costs can lead to changes in VSS. Since higher delay costs lead to higher second stage costs, even minor changes in flight control decisions may affect second stage costs significantly.

In Figure 6.2, we analyze the effect of different $\frac{r}{d}$ ratio values on VSS with more capacity limitations in the generated air traffic system. We observe that the behavior of VSS is very similar to larger capacity systems demonstrated in Figure 6.1. Therefore, we observe that having smaller capacities does not noticeably affect changes in VSS for these instances. However, when capacity is decreased more, VSS cannot be calculated due to infeasible subproblems.

We generate α_i's of weather events using $Uniform(0, 1 - c\_r)$, and c_r represents the maximum level that capacity levels can be affected by a weather event. We also analyze the change of VSS with respect to the maximum level that a weather event can affect capacity levels.

In Figure 6.3, we compare c_r and VSS and observe that, VSS is not significantly affected by c_r. This result is expected since capacities are generated more than adequate. In this setting, smaller c_r values lead to greater variation among scenarios, and this leads to infeasibilities in subproblems when the Expected Value Problem's first stage solution is used. Infeasibilites in subproblems correspond to flight cancellations and very large delay values in the air traffic system. As expected, these situations are very undesirable in real-life systems. Hence, solving stochastic model also has the advantage of providing the ability to

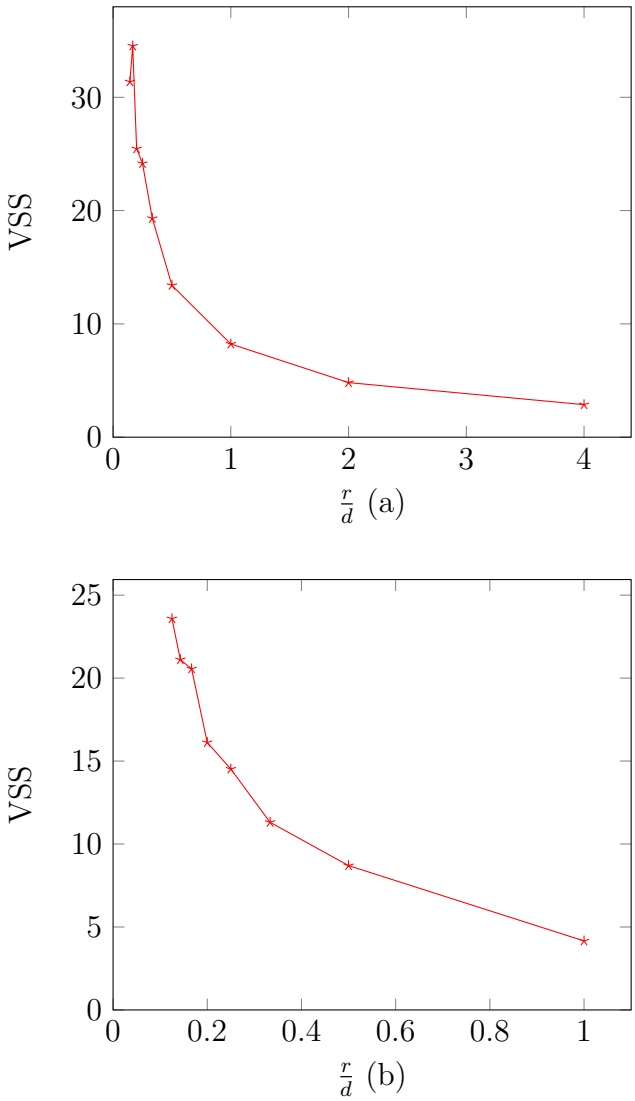


Figure 6.1: VSS for different values of route cost (r) to delay cost (d) ratio (more capacity). (a) 100 flights, (b) 200 flights

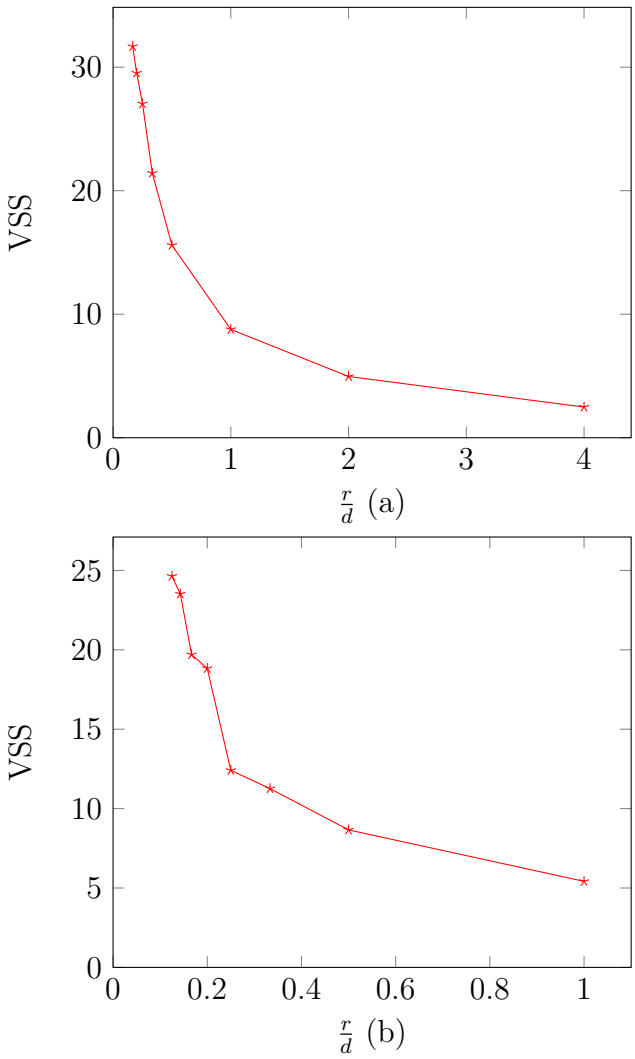


Figure 6.2: VSS for different values of route cost (r) to delay cost (d) ratio (less capacity). (a) 100 flights, (b) 200 flights

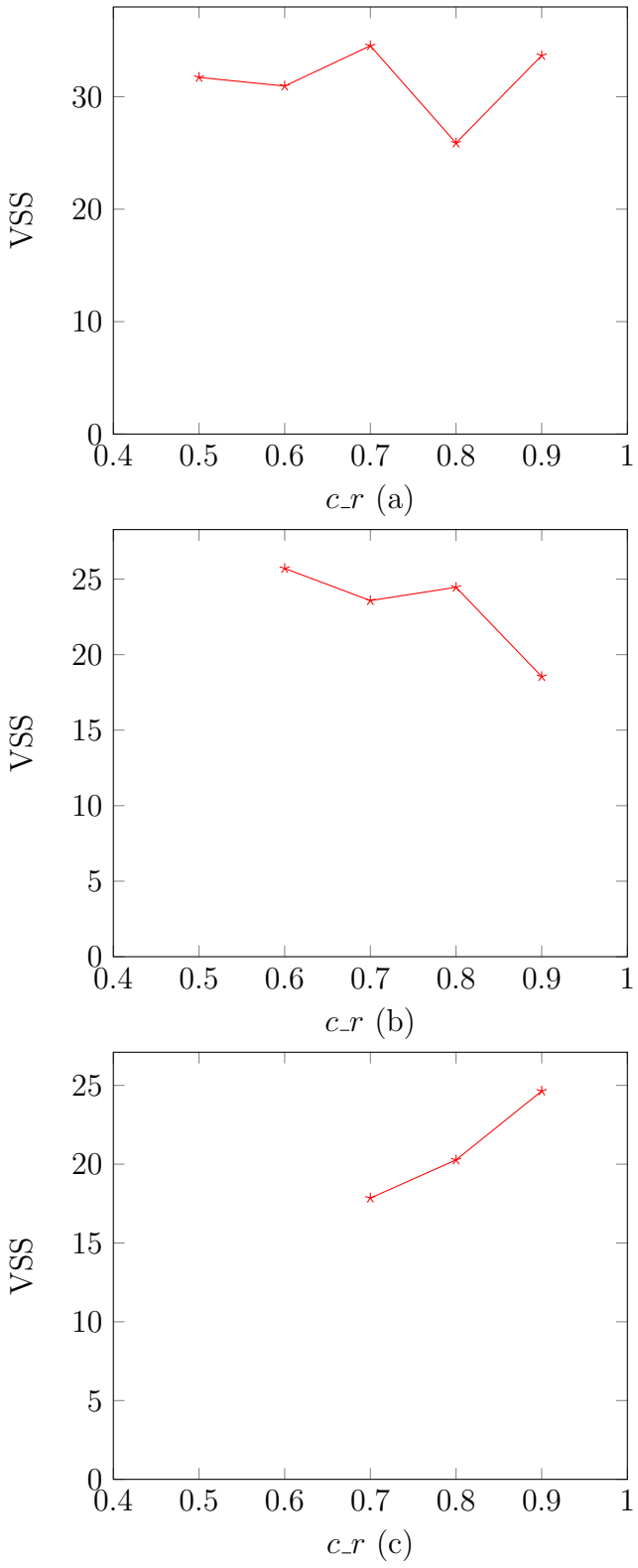


Figure 6.3: VSS for different values (c_r). (a) 100 flights and more capacity, (b) 200 flights and more capacity, (c) 200 flights and less capacity

avoid such situations. Details for VSS computations can be found in Appendix A.1.

6.3 Performance Comparison of Modified Partial Benders' and Improved Integer L-shaped

In this section, we compare performances of methodologies described in Sections 4.2 and 4.4 in terms of solution times and number of cuts added. We also discuss the implications of different parameters values on algorithm performances.

We report computational performances for different settings characterized by variation of connectivity level $\frac{|C|}{F}$ and unit delay cost. In the analysis, we refer Improved Integer L-shaped method as ILS and its combination with Partial Benders' as PBD.

In Table 6.1, we analyze different instances generated using $\frac{|C|}{F} = 10\%$ and $d = 0.25$. In this setting, we observe that as the number of flights gets larger, solution times of PBD (Partial Benders') outperform CPLEX without exception. In Figure 6.4, we illustrate computational advantages of using decomposition methods in larger problems since both ILS and PBD perform much better than CPLEX after 800 flights. Also, as expected with $d = 0.25$, VSS's are observed to be low when they are computable. When delay costs increase, VSS increases. However, it does not affect computational performance.

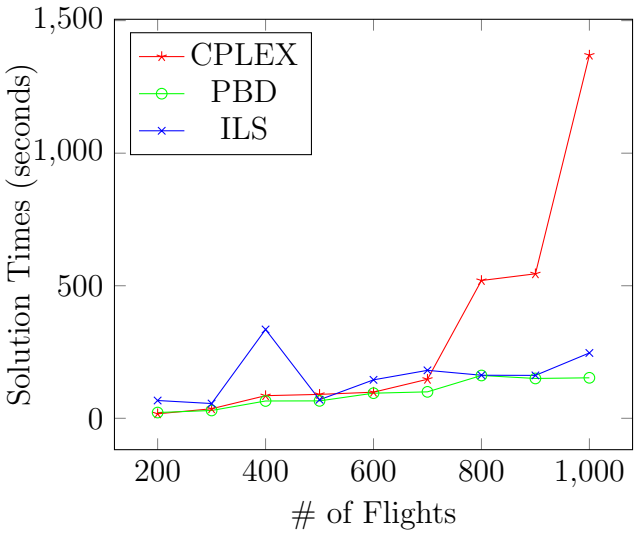


Figure 6.4: Solution times of problems with different number of flights in the setting of Table 6.1

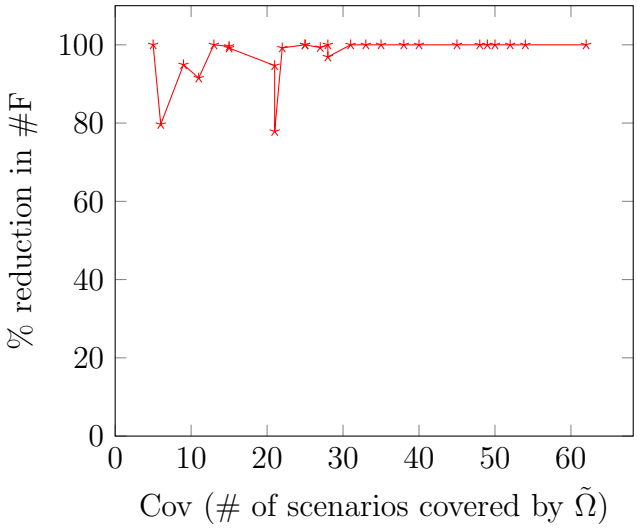


Figure 6.5: % reduction in feasibility cuts added with respect to number of scenarios covered by included scenarios (Cov) in the setting of Table 6.1

Instance Name	γ	$\lvert\tilde{\Omega}\rvert$	Cov	#F		#O		#S		# IO		Solution Time (seconds)			VSS
				PBD	ILS	PBD	ILS	PBD	ILS	PBD	ILS	PBD	CPLEX	ILS	
200_16_64_30_15_4_0.7_40	1	8	25	0	519	230	219	2	0	0	0	42.61	16.91	67.06	NA
	2	2	15	2		203		0		0		21.69			
	4	1	11	44		308		3		0		22.76			
300_24_64_30_15_4_0.7_40	1	16	28	0	242	96	136	0	0	0	0	33.92	35.96	55.7	1.24
	2	4	13	0		120		0		0		29.46			
	4	1	5	0		126		0		0		29.57			
400_24_64_30_15_4_0.7_40	1	25	50	0	1013	111	214	1	5	0	0	87.08	85.32	335.09	NA
	2	4	27	7		207		1		0		65.28			
	4	2	22	8		250		1		0		91.94			
500_32_64_30_15_4_0.7_40	2	4	45	0	27	72	78	0	0	0	0	70.95	90.16	69.97	0.18
	4	2	40	0		124		0		0		67.61			
	6	1	35	0		76		0		0		65.69			
600_32_64_30_15_4_0.7_40	1	23	54	0	226	82	192	0	0	0	0	116.52	98.45	145.03	NA
	2	2	28	7		129		0		0		94.69			
	6	1	21	12		127		0		0		110.97			
700_32_64_30_15_4_0.7_40	1	12	62	0	9	52	64	0	0	0	1	121.51	147.19	180.88	0.06
	3	3	52	0		61		0		0		100.64			
	6	2	49	0		62		0		8		99.73			
800_32_64_30_15_4_0.7_40	1	15	31	0	59	97	187	0	1	2	1	261.02	519.72	162.42	NA
	2	2	9	3		135		0		0		182.52			
	4	1	6	12		139		0		1		161.47			
900_32_64_30_15_4_0.7_40	2	4	25	0	122	111	118	1	0	2	0	163.53	544.97	161.84	NA
	4	2	21	27		124		0		0		157.63			
	8	1	15	1		127		0		5		150.24			
1000_32_64_30_15_4_0.7_40	1	12	48	0	301	156	187	0	0	0	0	397.32	1369.76	246.43	NA
	2	4	38	0		180		1		2		197.12			
	4	2	33	0		186		0		0		152.73			

Table 6.1: Computational aspects for varying number of flights using 64 scenarios when $\frac{|\mathcal{C}|}{F} = 10\%, \text{FS} = 4, \frac{\text{d}}{\text{r}} = 0.25, C_A = 20, C_S = 15, \text{c\_r} = 0.7, \text{NW} = 40, \mathcal{M} = 20$

In Table 6.1, we mark some VSS values as NA, meaning that one or more subproblems are infeasible using the first stage solution obtained from the Expected Value Problem. Practically, this would imply that there will be flight cancellations in those scenarios. This is caused by deficient capacities observed in some scenarios since NA values appear when $|\tilde{\Omega}|$ is small and Cov is large. This implies that there is a small number of scenarios having less capacity than many scenarios in all airspace elements. Although these cases are computationally advantageous for PBD, they are practically undesired in an air traffic system.

PBD also outperforms ILS in almost any case due to number of feasibility cuts reducing drastically. This is an expected result since this method is constructed specifically to achieve this property. In Figure 6.5, we analyze the percentage reduction in the number of feasibility cuts (denoted by #F in Table 6.1) with respect to number of scenarios covered by included scenarios (Cov). We observe that, as the number of covered scenarios increase, all of the feasibility cuts used in ILS are eliminated by Partial Benders'.

In Table 6.2, we analyze the computational performance with the number of scenarios. In this setting, we use six time periods in the first stage of the problem. As we increase the number of time periods in first stage, we observe that performance of PBD moves similarly to CPLEX. This can be attributed to Master Problem getting larger but not more informative as opposed to previous setting. We observe that, as the number of scenarios gets larger, CPLEX's performance is relatively better compared to the previous setting. Even though number of scenarios gets larger, the number of cuts generated by PBD does not move accordingly. This demonstrates that the cut reducing effect of PBD is not affected from number of scenarios.

Instance Name	γ	$\lvert\tilde{\Omega}\rvert$	Cov	#F		#O		#S		# IO		Solution Time (seconds)			VSS	Inf
				PBD	ILS	PBD	ILS	PBD	ILS	PBD	ILS	PBD	CPLEX	ILS		
300_24_16_30_15_6_0.7_40	0.5	5	16	0	381	11	32	0	0	0	0	2.27	8.29	45.65	19.76	0
	1	2	13	0		28		0		0		7.33				
	2	1	11	0		30		0		0		13.34				
300_24_20_30_15_6_0.7_40	0.5	5	20	0	430	30	26	0	0	0	0	8.79	8.76	51.65	17.31	0
	1	3	18	0		34		0		0		9.16				
	2	1	15	0		38		0		0		8.84				
300_24_24_30_15_6_0.7_40	1	4	13	20	648	80	59	0	0	0	0	16.37	13.85	79.41	NA	3
	2	2	11	14		45		0		0		14.24				
	4	1	9	17		46		0		0		13.53				
300_24_32_30_15_6_0.7_40	0.5	12	32	0	795	21	64	0	0	0	0	17.73	18.92	104.6	16.64	0
	1	5	25	0		54		0		0		15.32				
	4	1	20	0		124		0		0		17.93				
300_24_48_30_15_6_0.7_40	1	12	32	10	586	109	96	0	0	0	0	36.2	26.29	85.83	NA	6
	2	2	22	13		92		1		0		26.46				
	4	1	20	24		121		0		0		29.89				
300_24_64_30_15_6_0.7_40	1	15	41	0	255	89	106	0	2	0	0	48.07	45.11	81.83	NA	40
	2	4	25	0		124		2		0		43.22				
	4	2	20	18		115		0		0		56.59				
300_24_128_30_15_6_0.7_40	1	8	108	0	822	239	264	0	4	0	0	65.29	93.49	182.88	NA	88
	2	3	97	0		251		0		0		68.91				
	4	1	90	1		368		0		0		70.34				

Table 6.2: Computational aspects for increasing number of scenarios using 300 flights when $\frac{|\mathcal{C}|}{F} = 10\%, \text{FS} = 6, \frac{\text{d}}{\text{r}} = 6, C_A = 20, C_S = 15, \text{c\_r} = 0.7, \text{NW} = 40, \mathcal{M} = 20$

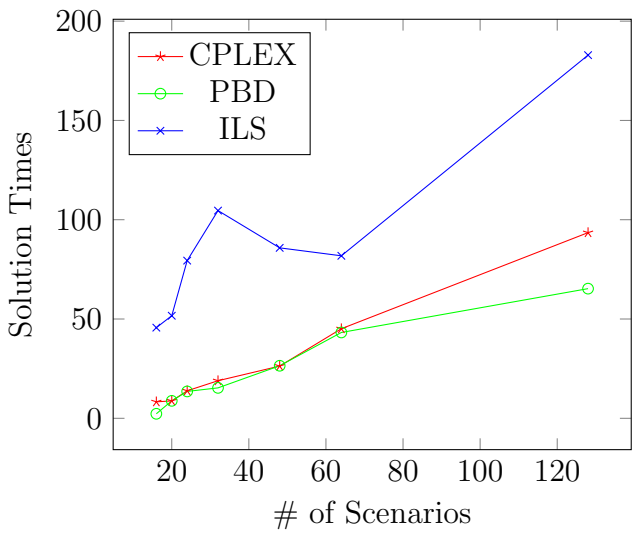


Figure 6.6: Solution times of problems with different number of scenarios in the setting of Table 6.2

In Figure 6.6, we observe that ILS is performing worse than CPLEX for any number of scenarios. This is due to the excessive number of subproblem evaluations. Also, since there are many subproblems, the master problem gets larger more quickly.

6.4 Experiments on CVaR-constrained Problem

In this section, we perform experiments on the accuracy and computational performance of the proposed solution methodology for the CVaR-constrained problem. We also analyze the change in the distribution of airline delays. As stated in Theorem 4.5.1, the approximation method for CVaR-constrained problem is a valid approximation, only when all the scenarios are covered, i.e., Cov $= |\Omega|$. This can be achieved by choosing $\gamma \in (0, 1)$ in the scenario selection problem. However, constructing $\tilde{\Omega}$ with such γ often results in computational difficulties since Master Problem gets much larger. Hence, we provide analyses addressing the issue both computationally and qualitatively. We elaborate on the trade-off between solution quality and computational performance using medium-sized instances, 500-600 flights, and 64 scenarios.

6.4.1 Analyses on Delay Distributions

In this section, we compare delay distributions for each airline and varying parameters of the problem. For each instance, we analyze resulting delays when different parameters are used. First, we analyze the case where selected scenarios "cover" all the scenarios in Ω. For illustrating the change in delay distributions, we solve the following versions of the (2SATFM) and (R2SATFM) problems for every instance generated:

- (2SATFM) without any delay restrictions, i.e. relaxing the constraint (3.19),
- (2SATFM) with maximum delay restrictions where $\mathcal{M} = 15$ and $\mathcal{M} = 10$,
- (R2SATFM) with following (α, β) values: $(0.95, 20), (0, 95, 15)$,
- Approximation of (R2SATFM) with the method described in Section 4.5, for previous (α, β) values.

500 Flights and 64 Scenarios: We generate an instance consisting of 500 flights, 30 sectors and 15 airports, and the resulting air traffic network is illustrated in Figure 6.7. We solve the aforementioned versions of (2SATFM) and (R2SATFM) problems and illustrate distributions of airline delays for this generated system.

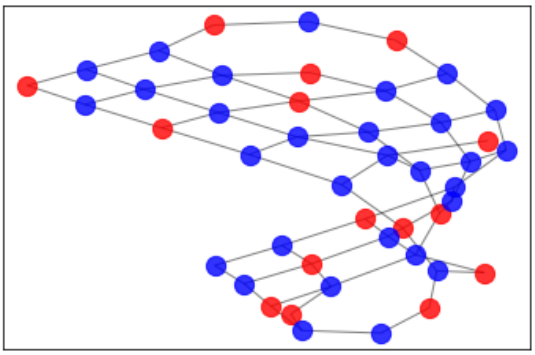


Figure 6.7: Air traffic network of generated instance

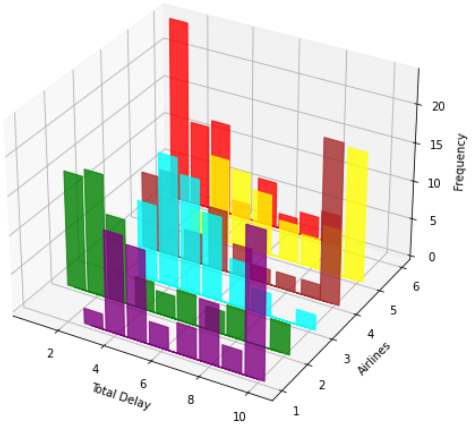
In Figure 6.8, we illustrate the distributions of total delay for each airline using histograms. In Figure 6.8c, histograms are provided when there is no restriction on delays. As expected, delays corresponding to some airlines are much higher compared to others. The underlying reason for this behavior may be that those airlines operate flights that can be used as compensators. Some flights that have more average scheduled departure times are more frequently operated by those airlines. When we check scheduled departure times for each airline, we see that the mean and standard deviation values of scheduled departure times for airlines 1,4, and 5 indicate less uncertainty. They have 3 "best" combinations of mean and standard deviation values.

In Figures 6.8a and 6.8b, delay distributions of (2SATFM) problem with two different maximum delay levels are provided. When the maximum allowed total airline delay, $\mathcal{M}$, is 10, we observe that for airlines 1,4, and 5, delays tend to cluster around the maximum delay level allowed. Since constraint (3.19) controls total airline delays by simply setting an upper bound, this is an expected result. In Figure 6.8b, we can observe a similar but milder cluster pattern, when $\mathcal{M} = 15$.

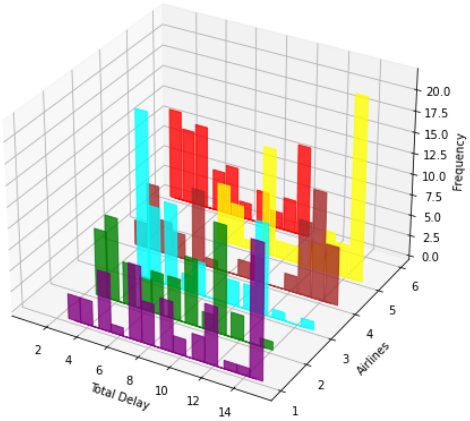
To avoid the cluster pattern as much as possible, we introduce CVaR constraints on total airline delays for each airline. In Figures 6.8d and 6.8e, we provide histograms of resulting delay distributions when CVaR constraints are used to control delays. In Figure 6.8d, we use $\alpha = 0.95, \beta = 20$ for CVaR. Which enforces that, the expectation of delays exceeding the 95% quantile cannot be larger than 20. As expected, we observe that cluster patterns are avoided when CVaR constraints are used. As observed in Figure 6.8e, the change in the behavior of delay clusters are very similar when $\beta = 15$ is used.

In Figure 6.9, we illustrate resulting total airline delays using box and whisker plots for different versions of (2SATFM) and (R2SATFM) problems. In addition to the observations from Figure 6.8, box and whiskers plots provide more informative pictures regarding the deviation and skewness of delay distributions.

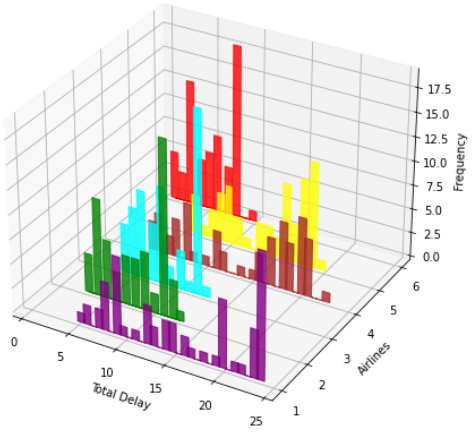
Figure 6.8: Histograms of total airline delays for 500 flights



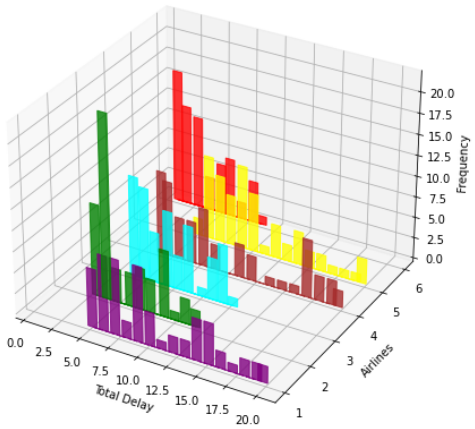
(a) $\mathcal{M} = 10$



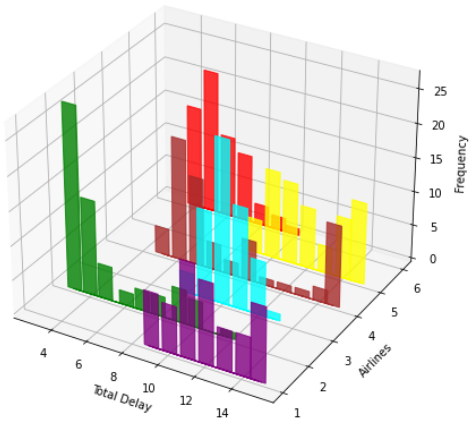
(b) $\mathcal{M} = 15$



(c) No restriction on delays

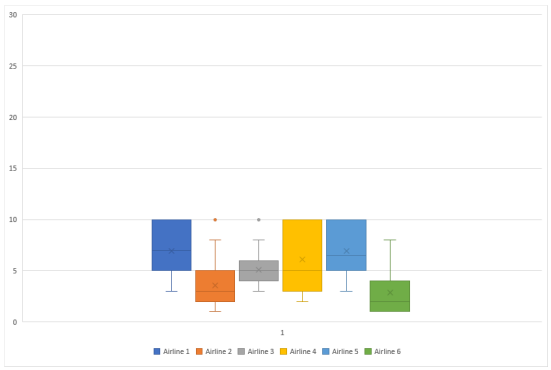


(d) CVaR constraints $\alpha = 0.95, \beta = 20$

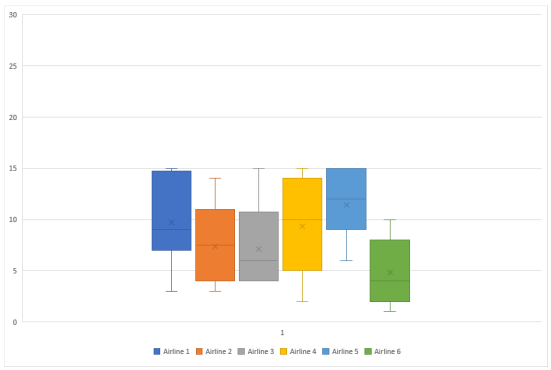


(e) CVaR constraints $\alpha = 0.95, \beta = 15$

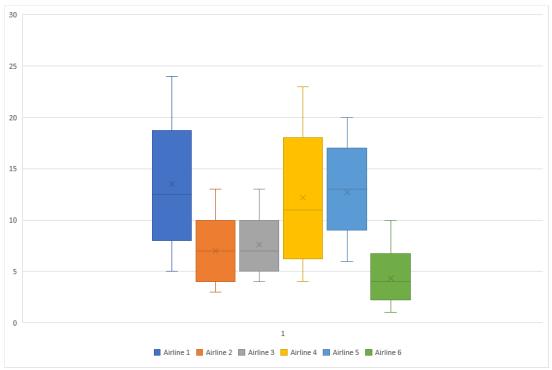
Figure 6.9: Box and whisker plots of total airline delays for 500 flights



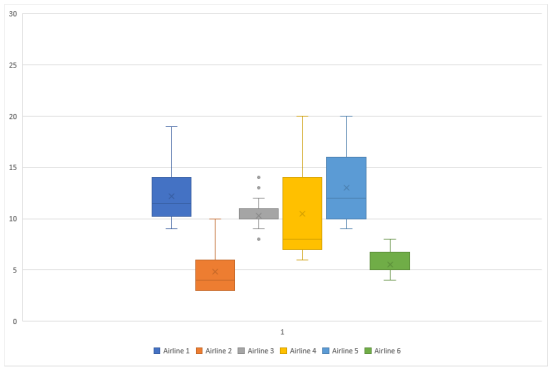
(a) $\mathcal{M} = 10$



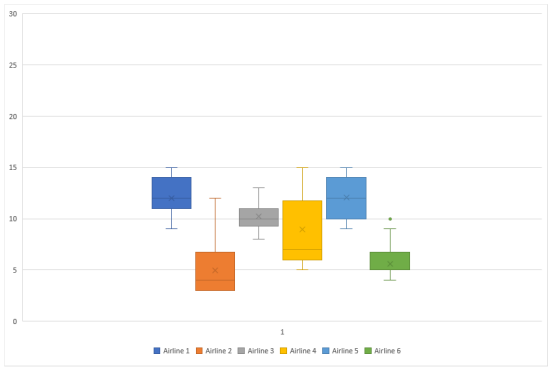
(b) $\mathcal{M} = 15$



(c) No restriction on delays



(d) CVaR constraints $\alpha = 0.95, \beta = 20$

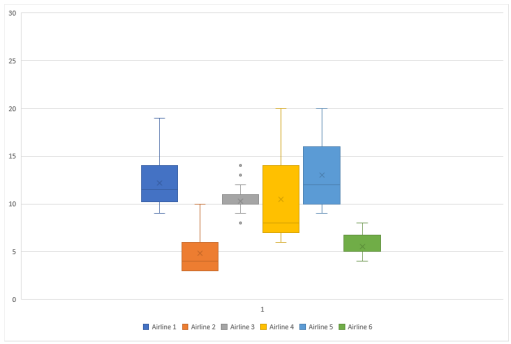


(e) CVaR constraints $\alpha = 0.95, \beta = 15$

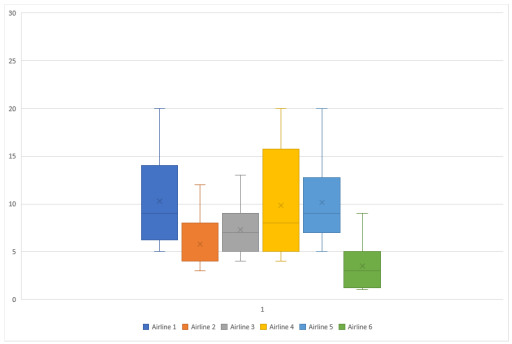
When Figures 6.9a and 6.9b are compared to Figures 6.9d and 6.9e, we can observe that CVaR constraints make delay distributions right skewed. Since maximum delay constraints result in clusters around boundaries, avoiding such cases

implies a more right skewed distribution. We can also observe that, variances of total airline delays get significantly smaller when CVaR constraints are used.

Note that, in Figures 6.8d, 6.8e, 6.9d and 6.9e, we use the exact solutions to get resulting delays. That is, we solve the extensive formulation as given in (R2SATFM). We also provide an analysis of total airline delay distributions, when the approximation method in Section 4.5 is used.



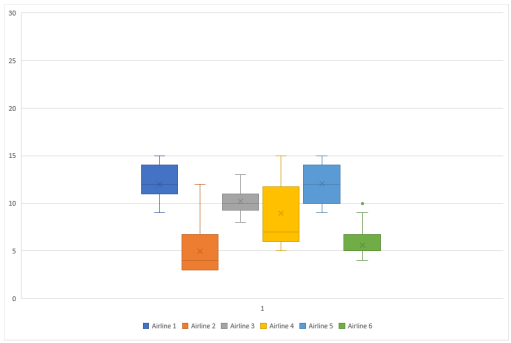
(a) CVaR with $\alpha = 0.95, \beta = 20$ (Exact Solution)



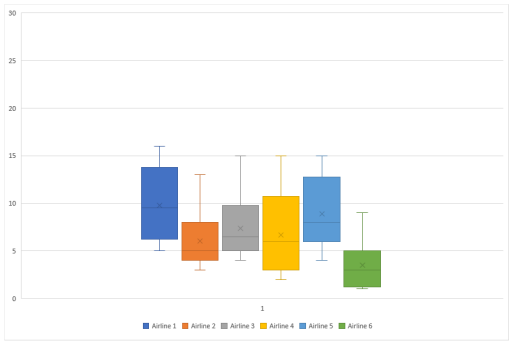
(b) CVaR with $\alpha = 0.95, \beta = 20$ (Approximate Solution)

Figure 6.10: Comparison of Approximate and Exact CVaR solutions for 500 flights ($\beta = 20$)

In Figure 6.10, we compare delay distributions resulting from exact and approximate solutions when $\beta = 20$. When the approximation method is used, delay distributions tend to have more variance than delays from the exact solution. This is an expected result since included scenarios provide an upper bound to non-included scenarios, and this upper bound is tight at the optimal solutions of subproblems. Hence, some scenarios result in more delays compared to the exact solution. This can also be interpreted as clusters of delays occurring around included scenario delays instead of boundaries.



(a) CVaR with $\alpha = 0.95, \beta = 15$ (Exact Solution)



(b) CVaR with $\alpha = 0.95, \beta = 15$ (Approximate Solution)

Figure 6.11: Comparison of Approximate and Exact CVaR solutions for 500 flights ($\beta = 15$)

In Figure 6.11, we observe a similar behavior. However, since β is smaller, some distributions change less significantly than the previous setting.

600 Flights and 64 Scenarios: We generate an instance consisting of 600 flights, 30 sectors, and 15 airports, and the resulting air traffic network is illustrated in Figure 6.12. We solve the aforementioned versions of (2SATFM) and (R2SATFM) problems and illustrate distributions of airline delays for this generated system.

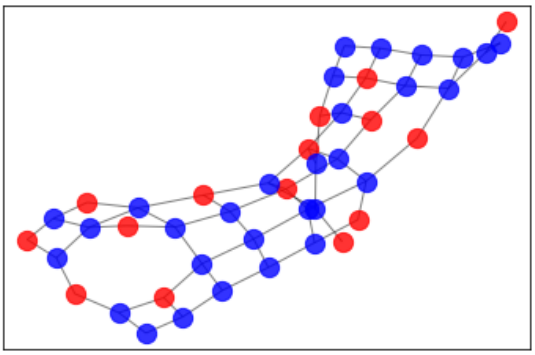
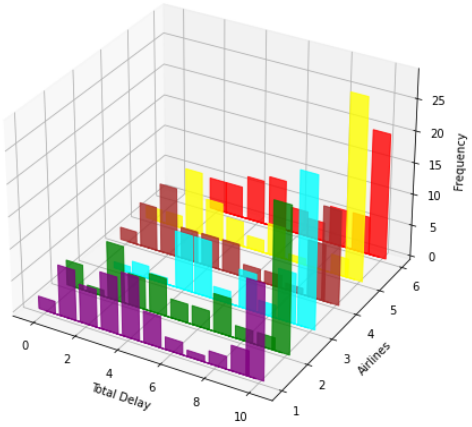
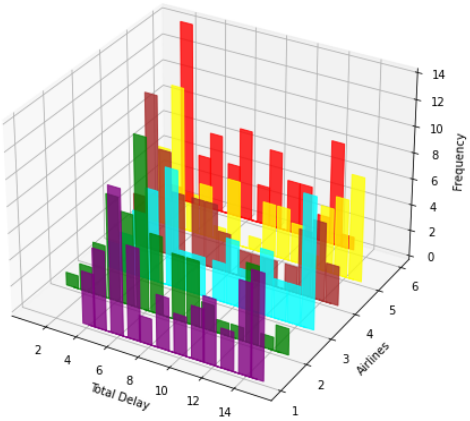


Figure 6.12: Air traffic network of generated instance

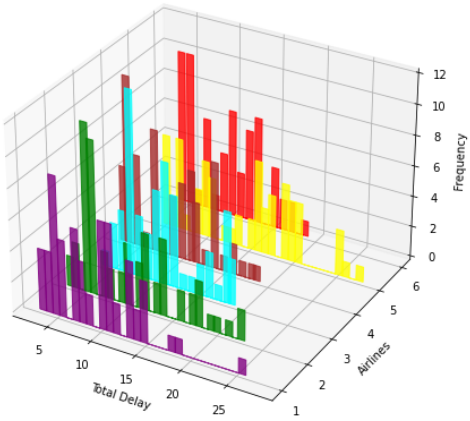
Figure 6.13: Histograms of total airline delays for 600 flights



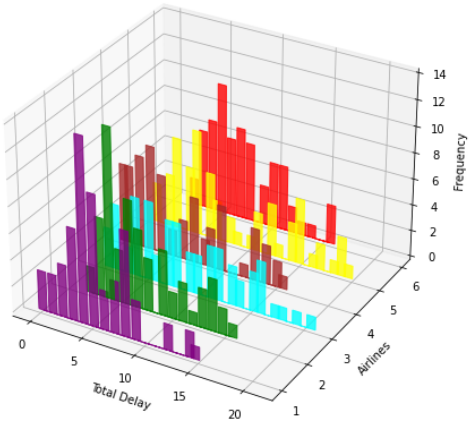
(a) $\mathcal{M} = 10$



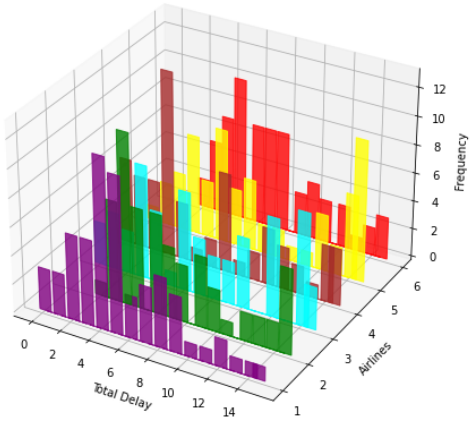
(b) $\mathcal{M} = 15$



(c) No restriction on delays



(d) CVaR constraints $\alpha = 0.95, \beta = 20$



(e) CVaR constraints $\alpha = 0.95, \beta = 15$

In Figure 6.13, we illustrate the distributions of total delay for each airline using histograms. In Figure 6.13c, histograms are provided when there is no restriction on delays. For this instance, there is no significant difference among total delays of different airlines. Similar to the 500 flight case, introducing maximum delay constraints cause delays to cluster around $\mathcal{M}$ values. This behavior can be observed in Figures 6.13a and 6.13a. As opposed to the previous example, delays resulting from maximum delay restrictions tend to cluster around non-boundary points, especially when $\mathcal{M} = 15$. Since similar clusters are also observed in no restrictions case, this is expected when the value of $\mathcal{M}$ is larger.

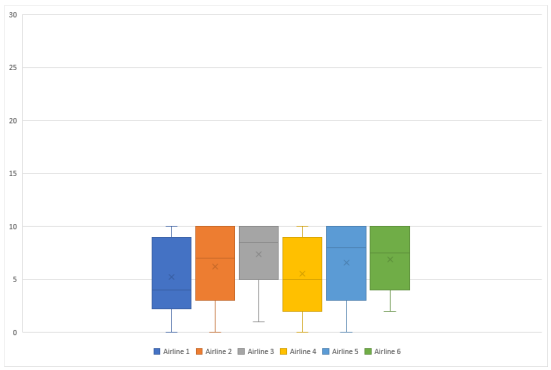
In Figures 6.13d and 6.13e, we provide histograms of resulting delay distributions when CVaR constraints are used to control delays. As intended with CVaR, clusters around extremes tend to get smaller. However, when $\beta = 20$, we observe that CVaR constraints are not tight enough to prevent inherent clusters, i.e., clusters from no restriction case.

In Figure 6.14, we illustrate resulting total airline delays using box and whisker plots for different versions of (2SATFM) and (R2SATFM) problems. In addition to the observations from Figure 6.13, box and whiskers plots provide more informative pictures regarding the deviation and skewness of delay distributions.

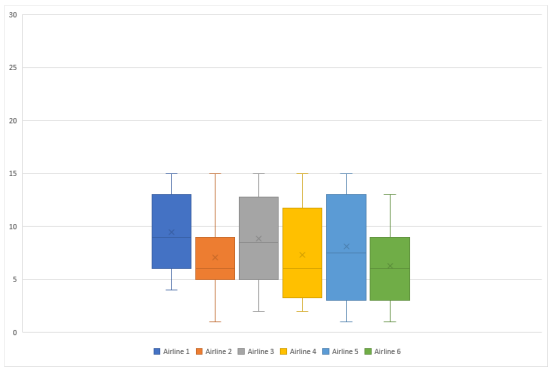
Figures 6.14d and 6.14e present more right-skewed distributions of total airline delays, similarly to 500 flight case. However, when $\beta = 20$, total delays resulting from CVaR-constrained solution, fail to have smaller variances. This behavior can be attributed to inherent clusters of delays, since it is observed only when $\beta = 20$.

Note that, in Figures 6.13d, 6.13e, 6.14d and 6.14e, we use the exact solutions to get resulting delays. That is, we solve the extensive formulation as given in (R2SATFM). We also provide an analysis of total airline delay distributions, when the approximation method in Section 4.5 is used.

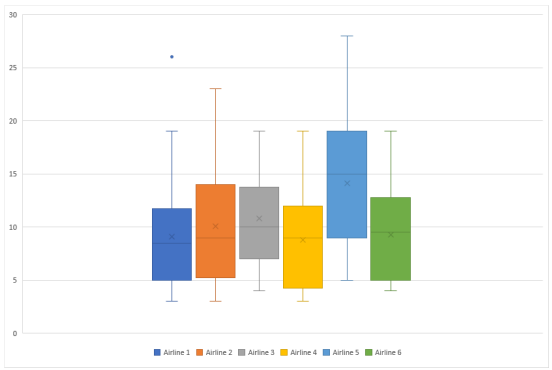
Figure 6.14: Box and whisker plots of total airline delays for 600 flights



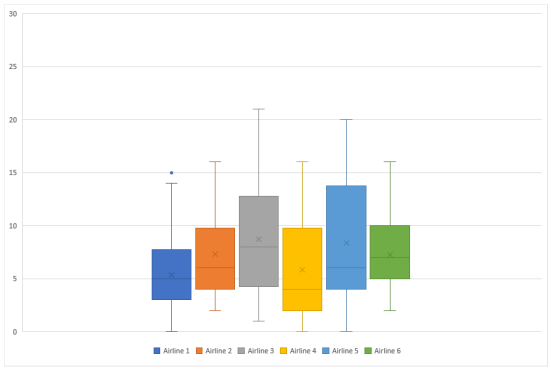
(a) $\mathcal{M} = 10$



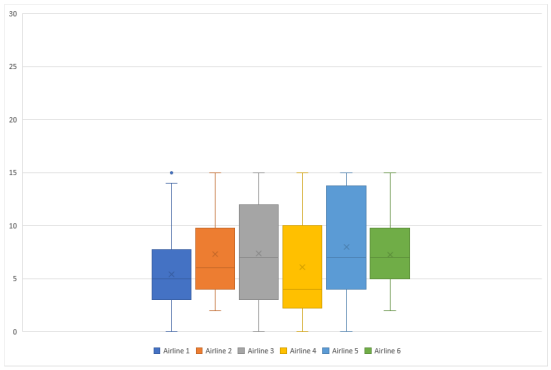
(b) $\mathcal{M} = 15$



(c) No restriction on delays



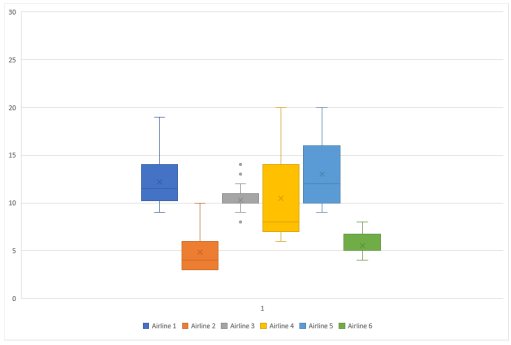
(d) CVaR constraints $\alpha = 0.95, \beta = 20$



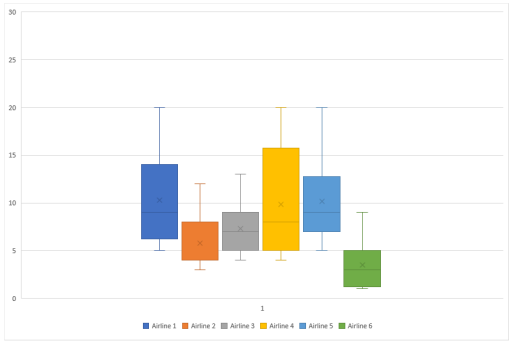
(e) CVaR constraints $\alpha = 0.95, \beta = 15$

In Figure 6.15, we compare delay distributions resulting from exact and approximate solutions when $\beta = 20$. When the approximation method is used, delay distributions tend to have more variance than delays from the exact solution,

similar to the 500 flights example. The underlying reason for this approximation error is also due to delay clusters around included scenario delays.



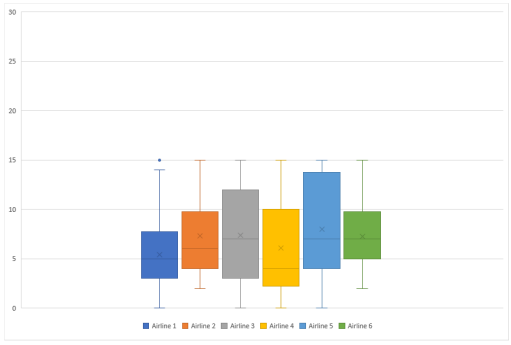
(a) CVaR with $\alpha = 0.95, \beta = 20$ (Exact Solution)



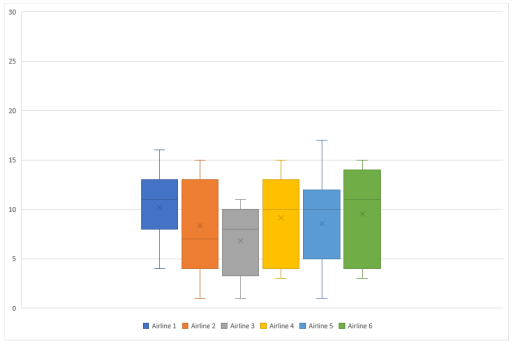
(b) CVaR with $\alpha = 0.95, \beta = 20$ (Approximate Solution)

Figure 6.15: Comparison of Approximate and Exact CVaR solutions for 500 flights ($\beta = 20$)

In Figure 6.16, we observe a similar behavior. However, since β is smaller, some distributions change less significantly than the previous setting.



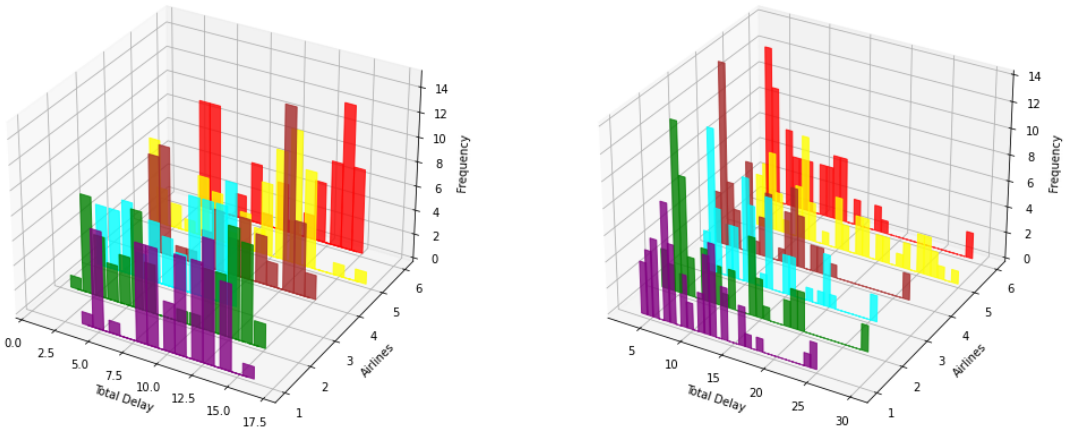
(a) CVaR with $\alpha = 0.95, \beta = 15$ (Exact Solution)



(b) CVaR with $\alpha = 0.95, \beta = 15$ (Approximate Solution)

Figure 6.16: Comparison of Approximate and Exact CVaR solutions for 500 flights ($\beta = 15$)

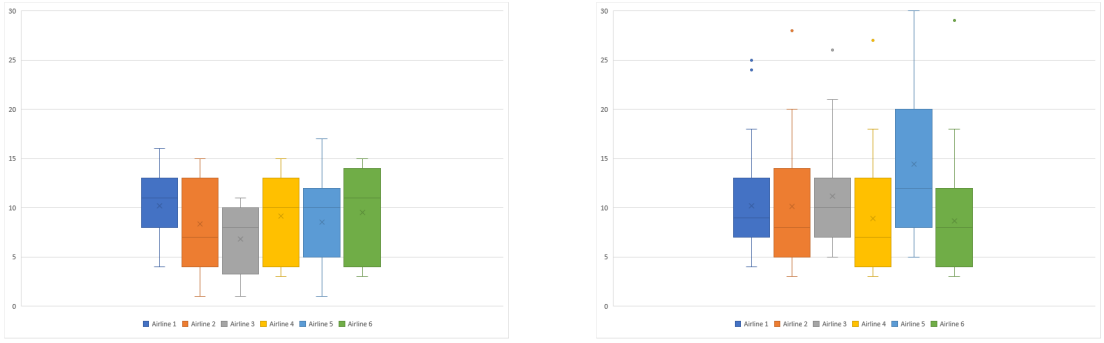
Note that, the approximation method used in Figures 6.15b and 6.16b, requires that, included scenarios must "cover" the whole sample space. This property enables us to control scenario delays for every scenario, using included delays. However, covering the whole space requires a large set of included scenarios, leading to more demanding computations. Hence, we use this example of 600 flights to demonstrate the effect of using a less than adequate number of scenarios.



(a) CVaR with $\alpha = 0.95, \beta = 15$ (Approximate Solution with Cov= 64)

(b) CVaR with $\alpha = 0.95, \beta = 15$ (Approximate Solution with Cov= 25)

Figure 6.17: Comparison of Approximate CVaR solutions for 600 flights with full and partial coverage of scenarios ($\beta = 15$)

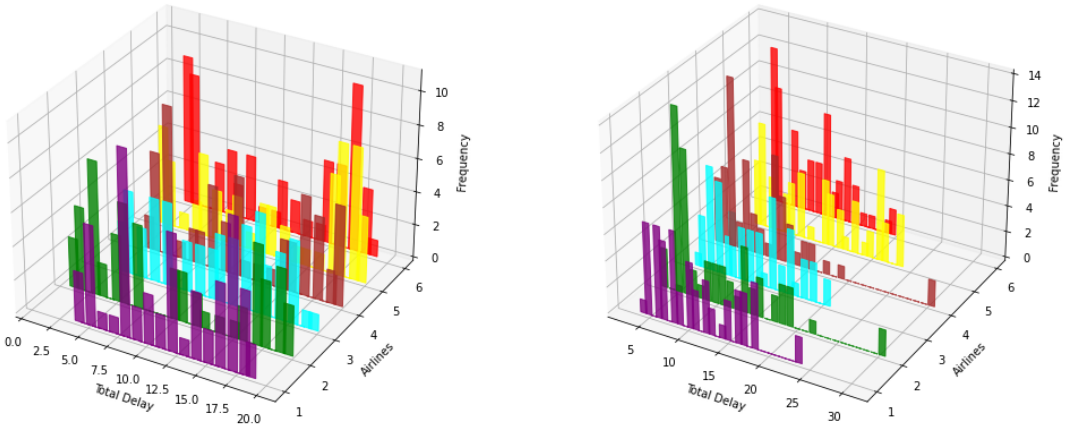


(a) CVaR with $\alpha = 0.95, \beta = 15$ (Approximate Solution with Cov= 64)

(b) CVaR with $\alpha = 0.95, \beta = 15$ (Approximate Solution with Cov= 25)

Figure 6.18: Comparison of Approximate CVaR solutions for 600 flights with full and partial coverage of scenarios ($\beta = 15$)

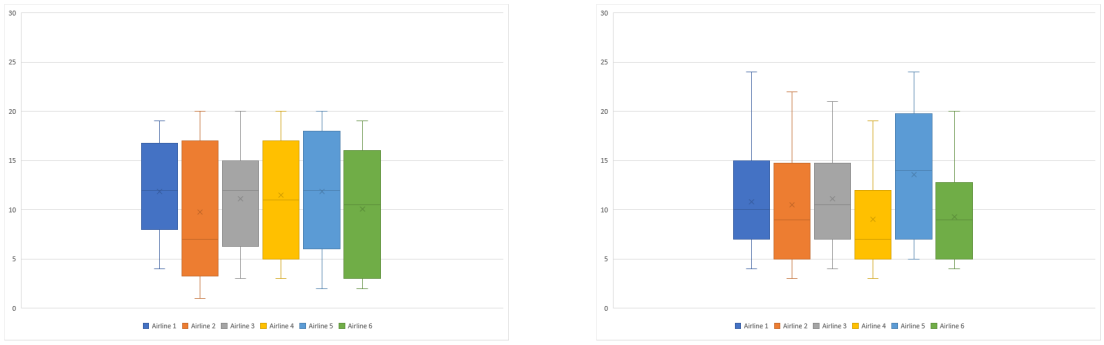
In Figure 6.17, one can observe that some scenarios that are not covered result in delays similar to not restricted version of the problem. This causes the right tails of delay distributions to get wider. However, as observed in Figure 6.18, the first and second quartiles remain similar in both fully and partially covered problems.



(a) CVaR with $\alpha = 0.95, \beta = 20$ (Approximate Solution with Cov= 64)

(b) CVaR with $\alpha = 0.95, \beta = 20$ (Approximate Solution with Cov= 25)

Figure 6.19: Comparison of Approximate CVaR solutions for 600 flights with full and partial coverage of scenarios ($\beta = 20$)



(a) CVaR with $\alpha = 0.95, \beta = 20$ (Approximate Solution with Cov= 64)

(b) CVaR with $\alpha = 0.95, \beta = 20$ (Approximate Solution with Cov= 25)

Figure 6.20: Comparison of Approximate CVaR solutions for 600 flights with full and partial coverage of scenarios ($\beta = 20$)

Similar behavior for approximation with partial coverage is also observed when $\beta = 20$, see Figures 6.19 and 6.20.

Note that, in this example, we include 43 scenarios in $\tilde{\Omega}$ to cover the sample space fully. However, for partially covering the sample space, we include only 4 scenarios to $\tilde{\Omega}$. Although partially covering the sample space leads to more error in approximations, it is observed to be computationally much more advantageous, see Table 6.3.

Problem Version	Solution Time
$\mathcal{M} = 15$	93.63
$\mathcal{M} = 10$	65.65
No restriction	72.71
CVaR Exact $\beta = 20$	268.83
CVaR Exact $\beta = 15$	257.36
CVaR Apx. $\beta = 20$ Full Coverage	100.41
CVaR Apx. $\beta = 15$ Full Coverage	131.86
CVaR Apx. $\beta = 20$ Partial Coverage	74.56
CVaR Apx. $\beta = 15$ Partial Coverage	75.49

Table 6.3: Solution times for different delay configurations for 600 flights

Note that, to solve for $\mathcal{M} = 15$, $\mathcal{M} = 10$ and "No restriction" versions of the (2SATFM) problem, we use PBD with $\gamma = 2$, corresponding to $|\tilde{\Omega}| = 4$ for this instance.

Chapter 7

Conclusions and Future Work

In this thesis, we provide mathematical programming models for the SATFM problem and propose different solution methodologies. We also provide a detailed description for generating realistic SATFM data using the effects of weather conditions. We focus on the version of SATFM problem where decisions can be made on take-off times and routes. This allows the control of demand-capacity imbalances in the air traffic network. As the underlying stochasticity is due to the capacities of airspace elements, resolving demand-capacity imbalances in this framework is crucial for practical purposes.

Proposed solution methodologies focus on two-stage stochastic programming formulation of SATFM problem, i.e. (2SATFM). To solve this problem, we combine two existing decomposition schemes: Improved Integer L-shaped and Partial Benders'. We provide extensive computational experiments to demonstrate the performance of the proposed method. We demonstrate that this method significantly outperforms state-of-the-art solver CPLEX up to a large number of flights, 1500. Moreover, we discover that this method can solve (2SATFM) problem at the point that CPLEX cannot, which is 1500 flights and 64 scenarios. Also, the performance of this method gets significantly better than CPLEX when the number of scenarios are increased up to 128.

Additionally, we consider Conditional Value-at-Risk (CVaR) constraints on total airline delays. To the best of our knowledge, CVaR constraints were never used in the ATFM domain. We also provide a novel approximation scheme for the CVaR-constrained version of (2SATFM) problem, i.e. (R2SATFM). Then, we discuss observed changes in total airline delay distributions when CVaR constraints are used. We demonstrate that using CVaR constraints to control delays leads to less restricted distributions of delays compared to using simple bounds on delays.

As for future research directions, one can consider the following:

- Generalizing and improving the approximation scheme for a wider class of problems,
- For (R2SATFM), modifying the approximation scheme by dynamically updating upper bounds on delays, using the "covering" relation defined across scenarios,
- Solving the multistage version of the two-stage problem by combining the proposed method and a rolling horizon heuristic.

Bibliography

[1] EUROCONTROL, "Performance review report on european air traffic management performance in 2019." https://www.eurocontrol.int/publication/performance-review-report-prr-2019, June 2020.

[2] A. T. A. Group, "Aviation: Benefits beyond borders." https://www.atag.org/component/attachments/attachments.html?id=961, October 2018.

[3] J. Smith, "Single european sky," Nov 2018.

[4] "About us eurocontrol." Available at https://www.eurocontrol.int/about-us.

[5] "National Airspace System." Available at https://www.faa.gov/air_traffic/nas.

[6] G. Laporte and F. V. Louveaux, "The integer l-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, vol. 13, no. 3, pp. 133 – 142, 1993.

[7] G. Angulo, S. Ahmed, and S. S. Dey, "Improving the integer l-shaped method," *INFORMS Journal on Computing*, vol. 28, no. 3, pp. 483–499, 2016.

[8] T. G. Crainic, M. Hewitt, F. Maggioni, and W. Rei, "Partial benders decomposition: General methodology and application to stochastic network design," *Transportation Science*, vol. 55, no. 2, pp. 414–435, 2021.

[9] A. R. Odoni, "The flow management problem in air traffic control," in *Flow Control of Congested Networks* (A. R. Odoni, L. Bianco, and G. Szegö, eds.), (Berlin, Heidelberg), pp. 269–288, Springer Berlin Heidelberg, 1987.

[10] T. W. M. Vossen, R. Hoffman, and A. Mukherjee, *Air Traffic Flow Management*, pp. 385–453. Boston, MA: Springer US, 2012.

[11] R. Shone, K. Glazebrook, and K. G. Zografos, "Applications of stochastic modeling in air traffic management: Methods, challenges and opportunities for solving air traffic problems under uncertainty," *European Journal of Operational Research*, vol. 292, no. 1, pp. 1–26, 2021.

[12] D. Bertsimas and S. Stock Patterson, "The air traffic flow management problem with enroute capacities," *Operations Research*, vol. 46, pp. 406–422, 1998.

[13] G. Lulli and A. Odoni, "The european air traffic flow management problem," *Transportation Science*, vol. 41, no. 4, pp. 1–13, 2007.

[14] D. Bertsimas, G. Lulli, and A. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Operations Research*, vol. 59, pp. 211–227, January 2011.

[15] A. Agustín, A. Alonso-Ayuso, L. Escudero, and C. Pizarro, "On air traffic flow management with rerouting. part i: Deterministic case," *European Journal of Operational Research*, vol. 219, no. 1, pp. 156–166, 2012.

[16] G. Andreatta and G. Romanin-Jacur, "Aircraft flow management under congestion," *Transportation Science*, vol. 21, no. 4, pp. 249–253, 1987.

[17] O. Richetta and A. Odoni, "Solving optimally the static ground-holding policy problem in air traffic control," *Transportation Science*, vol. 27, no. 3, pp. 228–238, 1993.

[18] M. O. Ball, R. Hoffman, A. R. Odoni, and R. Rifkin, "A stochastic integer program with dual network structure and its application to the ground-holding problem," *Operations Research*, vol. 51, no. 1, pp. 167–171, 2003.

[19] B. Kotnyek and O. Richetta, "Equitable models for the stochastic ground-holding problem under collaborative decision making.," *Transportation Science*, vol. 40, pp. 133–146, 2006.

[20] A. Mukherjee and M. Hansen, "A dynamic stochastic model for the single airport ground holding problem," *Transportation Science*, vol. 41, pp. 444–456, November 2007.

[21] P. Vranas, D. Bertsimas, and A. Odoni, "The multi-airport ground holding problem in air traffic control," *Operations Research*, vol. 42, pp. 249–261, 1994.

[22] G. Andreatta, P. Dell'Olmo, and G. Lulli, "An aggregate stochastic programming model for air traffic flow management," *European Journal of Operational Research*, vol. 215, no. 3, pp. 697–704, 2011.

[23] A. Alonso, L. F. Escudero, and M. T. Ortuno, "A stochastic 0-1 program based approach for the air traffic flow management problem," *European Journal of Operational Research*, no. 120, pp. 47–62, 2000.

[24] A. Agustín, A. Alonso-Ayuso, L. Escudero, and C. Pizarro, "On air traffic flow management with rerouting. part II: Stochastic case," *European Journal of Operational Research*, vol. 219, no. 1, pp. 167–177, 2012.

[25] D. Bertsimas and S. Gupta, "Multistage air traffic flow management under capacity uncertainty: A robust and adaptive optimization approach," 2012. submitted.

[26] A. Nilim and L. E. Ghaoui, "Algorithms for air traffic flow management under stochastic environments," in *Proceedings of the 2004 American Control Conference*, pp. 3429–3434, July 2004.

[27] A. Nilim and L. E. Ghaoui, "Robust control of markov decision processes with uncertain transition matrices," *Operations Research*, vol. 53, no. 5, pp. 780–798, 2005.

[28] S. Ichoua, "A scenario-based approach for the air traffic flow management problem with stochastic capacities," *International Journal of Mechanical,*

Aerospace, Industrial and Mechatronics Engineering, vol. 7, no. 8, pp. 596–598, 2013.

[29] Y.-H. Chang, S. Solak, J.-P. B. Clarke, and E. L. Johnson, "Models for single-sector stochastic air traffic flow management under reduced airspace capacity," *Journal of the Operational Research Society*, vol. 67, no. 1, pp. 54–67, 2016.

[30] L. Corolli, G. Lulli, L. Ntaimo, and S. Venkatachalam, "A two-stage stochastic integer programming model for air traffic flow management," *IMA Journal of Management Mathematics*, vol. 28, pp. 19–40, 05 2015.

[31] H. Balakrishnan and B. Chandran, "Optimal large-scale air traffic flow management," 2014. Available at http://web.mit.edu/hamsa/www/publications.html.

[32] J. Chen, L. Chen, and D. Sun, "Air traffic flow management under uncertainty using chance-constrained optimization," *Transportation Research Part B: Methodological*, vol. 102, pp. 124–141, 2017.

[33] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, pp. 21–41, 2000.

[34] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, "Coherent measures of risk," *Mathematical Finance*, vol. 9, no. 3, pp. 203–228, 1999.

[35] J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numer. Math.*, vol. 4, p. 238–252, Dec. 1962.

[36] R. M. V. Slyke and R. Wets, "L-shaped linear programs with applications to optimal control and stochastic programming," *SIAM Journal on Applied Mathematics*, vol. 17, no. 4, pp. 638–663, 1969.

[37] J. R. Birge and F. V. Louveaux, "A multicut algorithm for two-stage stochastic linear programs," *European Journal of Operational Research*, vol. 34, no. 3, pp. 384–392, 1988.

[38] EUROCONTROL, "Technical note on en route capacity." https://www.eurocontrol.int/sites/default/files/2020-12/eurocontrol-prc-technical-note-en-route-capacity-11122020.pdf, December 2020.

Appendix A

Details of Computational Experiments

A.1 Details of VSS Computations

r	d	C_A	C_S	$\|\mathcal{C}\|$	M	c_r	VSS	Avg. VSS
1	7	20	15	10	20	0.7	35.55	31.37
							29.5	
							29.7	
							28.7	
							33.4	
1	6	20	15	10	20	0.7	39.59	34.534
							35.55	
							28.86	
							34.38	
							34.29	
1	5	20	15	10	20	0.7	23.57	25.46
							21.95	
							30.94	
							21.56	
							29.28	
1	4	20	15	10	20	0.7	22.41	24.158
							25.27	
							23.51	
							23.88	
							25.72	
1	3	20	15	10	20	0.7	16.76	19.312
							18.73	
							20.31	
							20.98	
							19.78	
1	2	20	15	10	20	0.7	14.12	13.4
							11.19	
							10.68	
							16.38	
							14.63	
1	1	20	15	10	20	0.7	9.2	8.224
							8.34	
							7.66	
							7.27	
							8.65	

Table A.1: Value of Stochastic Solution for F=100 A= 15 S= 30 NW=40 T=16 C=10 M=20 each parameter is tested for 5 different data randomly generated and high capacities are generated

r	d	C_A	C_S	$\|\mathcal{C}\|$	M	c_r	VSS	Avg. VSS
1	6	10	8	10	20	0.7	25.85	31.676
							28.9	
							28.45	
							38.7	
							36.48	
1	5	10	8	10	20	0.7	31.57	29.528
							29.75	
							24.07	
							29.16	
							33.09	
1	4	10	8	10	20	0.7	27.85	27.038
							27.82	
							25.77	
							29.23	
							24.52	
1	3	10	8	10	20	0.7	19.58	21.422
							24.76	
							24.45	
							15.95	
							22.37	
1	2	10	8	10	20	0.7	18.03	15.59
							15.68	
							15.64	
							14.63	
							13.97	
1	1	10	8	10	20	0.7	8.97	8.772
							8.47	
							8.81	
							7.99	
							9.62	

Table A.2: Value of Stochastic Solution for F=100 A= 15 S= 30 NW=40 T=16 C=10 M=20 each parameter is tested for 5 different data randomly generated and medium capacities are generated

r	d	C_A	C_S	$\|\mathcal{C}\|$	M	c_r	VSS	Avg. VSS
1	8	20	15	40	20	0.7	22.47	23.58333
							22.39	
							25.89	
1	7	20	15	40	20	0.7	17.8	21.11333
							24.52	
							21.02	
1	6	20	15	40	20	0.7	12.66	20.56667
							25.33	
							23.71	
1	5	20	15	40	20	0.7	16.41	16.12333
							15.21	
							16.75	
1	4	20	15	40	20	0.7	11.94	14.52333
							16.43	
							15.2	
1	3	20	15	40	20	0.7	10.36	11.30667
							12.83	
							10.73	
1	2	20	15	40	20	0.7	5.58	8.703333
							10.24	
							10.29	
1	1	20	15	40	20	0.7	4.55	4.16
							4.64	
							3.29	

Table A.3: Value of Stochastic Solution for F=200 A= 15 S= 30 NW=40 T=16 C=10 M=20 each parameter is tested for 3 different data randomly generated and high capacities are generated

r	d	C_A	C_S	$\|\mathcal{C}\|$	M	c_r	VSS	Avg. VSS
1	8	15	10	40	20	0.7	24.85	24.64667
							19.82	
							29.27	
1	7	15	10	40	20	0.7	22.9	23.51667
							27.08	
							20.57	
1	6	15	10	40	20	0.7	15.81	19.7
							22.16	
							21.13	
1	5	15	10	40	20	0.7	19.63	18.83
							19.94	
							16.92	
1	4	15	10	40	20	0.7	12.57	12.41
							11.92	
							12.74	
1	3	15	10	40	20	0.7	13.5	11.26667
							10.72	
							9.58	
1	2	15	10	40	20	0.7	8.09	8.66
							7.48	
							10.41	
1	1	15	10	40	20	0.7	5.82	5.423333
							4.99	
							5.46	

Table A.4: Value of Stochastic Solution for F=200 A= 15 S= 30 NW=40 T=16 C=10 M=20 each parameter is tested for 3 different data randomly generated and medium capacities are generated