ABSTRACT

Title of document: COORDINATED AND ROBUST AVIATION NETWORK RESOURCE ALLOCATION

Andrew M. Churchill, Doctor of Philosophy, 2010

Directed by: David J. Lovell, Associate Professor, Department of Civil & Environmental Engineering

In the United States, flight operators may schedule flights to most airports at whatever time best achieves their objectives. However, during some time periods, both at airports and in the airspace, these freely-developed schedules may become infeasible because weather or other factors reduce capacity. A plan must then be implemented to mitigate this congestion safely, efficiently, and equitably. Current planning processes treat each congested resource independently, applying various rules to increase interoperation times sufficiently to match the reduced capacity. However, several resources are occasionally congested simultaneously, and ignoring possible dependencies may yield infeasible allocations for flights using multiple resources.

In this dissertation, this problem of developing coordinated flight-slot allocations for multiple congested resources is considered from several perspectives. First, a linear optimization model is developed. It is demonstrated that optimally minimizing flight arrival delays induces an increasing bias against flights using multiple resources. However, the resulting allocations reduce overall arrival delay, as compared to the infeasible independent allocations, and to current operational practice. The analytic properties of the model are used to develop a rule-based heuristic for allocating capacity that achieves comparable aggregate results. Alternatively, minimizing delay assigned at all resources is considered, and this objective is shown to mimic the flights' original schedule order.

Recognizing that minimizing arrival delays is attractive because of its tangible impact on system performance, variations to the original optimization model are proposed that constrain the worst-case performance of any individual user. Several different constraints and cost-based approaches are considered, all of which are successful to varying degrees in limiting inequities.

Finally, the model is reformulated to consider uncertainty in capacity. This adds considerable complexity to the formulation, and introduces practical difficulties in identifying joint probability distributions for the capacity outcomes at each resource. However, this new model is successful in developing more robust flight-slot allocations that enable quick responses to capacity variations.

Each of the optimization models and heuristics presented here are tested on a realistic case study. The problem studied and the approaches employed represent an important middle ground in air traffic flow management research between single resource models and comprehensive ones.

COORDINATED AND ROBUST AVIATION NETWORK RESOURCE ALLOCATION

By

Andrew M. Churchill

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2010

Advisory Committee:
Professor David J. Lovell, Chair
Professor Michael O. Ball
Professor Elise Miller-Hooks
Professor Paul Schonfeld
Dr. Robert Hoffman

Dedication

To Alison, with love.

Acknowledgements

My years in graduate school have been both challenging and highly fulfilling. They have been enriched by many friendly faces that have helped me along the way.

Few students have the privilege (or punishment) of having the same mentor and advisor for nine years, as I have with Dave Lovell. I like to imagine that my development from an 18 year old expert on everything to whatever I am now provides him no small amusement. Every student should be so lucky as to work with someone who cares so deeply about students. I am forever grateful for all the guidance and advice he has provided me, and for the friendship and the excellent trips that we have shared over the years.

Michael Ball and Robert Hoffman have carefully guided both my professional and personal development throughout my time in graduate school. I appreciate all that they have taught me about aviation, operations research, and more. Among other contributions, Mike has always provided me with invaluable guidance on my research. Bob has always brought an outside view to my graduate school experience that has helped me keep perspective. I gratefully acknowledge my other committee members, Elise Miller-Hooks and Paul Schonfeld, not only for their assistance with this dissertation, but also as excellent teachers I have relied on throughout graduate school and as shining examples of successful faculty.

In pursuing my degree, I have been fortunate to have received financial support, through Dave and Mike's good graces, from the Federal Aviation Administration, the National Aeronautics and Space Administration, the United States Department of Transportation, the A. James Clark School of Engineering, and the Graduate School at the University of Maryland.

I appreciate the camaraderie, discussion, and support of the many people with whom I've shared an office over the years. In no particular order, I thank Kleoniki Vlachou, Charles Glover, Shin-Lai Tien, Moein Ganji, Nasim Vakili, Bargava

Subramanian, Ming Zhong, Yufeng Tu, and Kennis Chan. Although we've never shared a physical office for more than a few days at a time, I am eternally grateful to Megan Smirti Ryerson as both a friend and a colleague for all that she has been to me during graduate school.

Without the foundations my parents Robin and Robert Churchill laid for me and the structure they have provided along the way, none of this would have been possible. Every day I strive to be as strong, hardworking, and dedicated as they are.

To my ~~friend~~ ~~girlfriend~~ ~~fiancée~~ wife Alison I dedicate this dissertation. To her, I owe the strongest acknowledgements, the deepest gratitude, and the most favors. I now know that she is the reason I came to Maryland for graduate school, the reason I stayed, and the reason I have finally finished.

Table of Contents

List of Tables

List of Figures

1. Introduction

In the United States, flight operators are free to schedule and operate flights to most airports at whatever time and by whatever route best achieves their objectives. However, during congested time periods, both at airports and in the airspace, these freely-developed schedules may become infeasible. At that time, the system operator, the Federal Aviation Administration, must develop a plan to mitigate this congestion safely and efficiently.

At some times, several resources are congested simultaneously. Current planning processes treat each of these resources independently, and apply various rules and processes to increase interoperation times sufficiently to match the reduced available capacity. This assumption of independence may yield infeasible capacity allocations in the case that some flights are using multiple resources.

For example, a flight may travel through a region of airspace congested because of weather before arriving at an airport that has closed one of its runways. Each of these disruptions necessitates a systematic plan for preventing local congestion, but it is likely that the plans developed independently will not be compatible with one another for this flight. Previous research has either not addressed this coordination problem, or has taken a comprehensive view, in which assumptions about operator rights are disregarded, that allocates capacity at every airport and division or airspace, regardless of expected congestion.

This dissertation focuses on strategies and models to develop coordinated slot allocations at multiple resources without the assumption of independence, while considering only congested and connected resources. In the next section, the capacity allocation principles and systems used in practice, and those proposed by researchers, are outlined. Then, the problem of coordination between capacity allocation programs is described in greater detail.

1.1 Capacity rationing principles and systems

Congested resources in the airspace system require intervention to maintain safety. In the United States, the FAA is provided the statutory authority to make these interventions and operate the air traffic system so as "to prevent a collision between aircraft operating in the system and to organize and expedite the flow of traffic, and to provide support for National Security and Homeland Defense." (Federal Aviation Administration 2008). The means to achieve this objective can be loosely categorized according to their time scales: air traffic controls reflects tactical actions made primarily for safety, while air traffic flow management (ATFM) reflects strategic interests made to maximize and balance efficiency and equity in the system.

One of the primary roles of the ATFM system is to allocate capacity at congested resources. These processes proceed under the assumption that congested resources, and the expected duration of the demand-capacity imbalance, have been identified by some external process. Given this information, they develop a detailed plan to match projected demand with a resource's expected capacity. It is important to note that these allocations are developed simply as plans and that flight operators have considerable leeway, within their share of the allocation, in managing their own operations. The mechanisms, operational systems, and advanced research that inform this process are described in this section. In a broader sense, this problem represents one of a general class of resource allocation models.

1.1.1 *Principles*

Underlying virtually all ATM tools are rationing principles, because, at their core, these are all procedures to ration scarce resources. These rationing principles are important, and will be used throughout the remainder of this research. Each rationing principle begins with some notional baseline from which the procedure begins. Several pieces of data about each flight provide good candidates as the starting point for allocating priority, including both a flight's scheduled time and currently projected time at each congested resource they encounter. Scheduled time is used for several reasons outlined below.

Flight operators publish schedules indicating only departure and arrival times months in advance.   Aircraft trajectories are well understood, enabling a reasonably accurate projection of the flight's position between origin and destination. This allows for inference of the "scheduled" time to arrive at some en route resource. Airline schedules are published long in advance, which effectively prohibits gaming to gain advantage in any individual capacity rationing program on the day of operation. However, there are many non-scheduled flights, including both business and general aviation, which present unique challenges in using schedule time as a baseline, as they lack this data.

Alternatively, projected time may be used for each flight. Before departure, all users operating in airspace with which this work is concerned must file flight plans indicating their proposed flight path. In addition, flight operators share with the FAA information about planned departure and arrival times. These may differ from scheduled times as a result of a variety of factors occurring on the particular day in question. Based on this information, and using the same projection methods as for scheduled times, a time-varying projection of flight position can be developed, yielding projected arrival times at each resource. This baseline is more complete than schedule because all flights can be projected based on current data. However, it provides an incentive for users to falsify information to gain advantage. For example, users may gain higher priority by providing false information about flights that they know will be delayed.

Of these two baselines, schedule time is generally used. This is primarily to avoid the possibility of gaming by users by encouraging truth-telling. Schedules are, by their nature, long-term and strategic in nature. Their development is an intensive process, and in a network environment such as that used by air carriers, changes in one location have a system-wide impact. All this argues that users will find it difficult to game the system with schedules. In the original implementations of airport capacity rationing used in the United States, projected time was used, and this approach encountered considerable difficulties that led to the introduction of scheduled arrival time as the standard for rationing. The problems with this approach are described in greater detail in (Vossen and Ball 2006).

Thus, the most basic procedure for rationing capacity in the U.S. airspace is known as Ration By Schedule (RBS). When rationing becomes necessary under reduced capacity, flights are prioritized according to their published arrival times. This process works, at a basic level, by simply stretching out the schedule of flights to match some reduced capacity. A trivial example of this is shown in Figure 1-1, wherein ten flights are nominally scheduled with two minute interarrival times. Due to some reduction in capacity, however, interarrival times must be increased to four minutes. Flights must be delayed to meet this new target. The first flight receives no delay, while subsequent flights receive linearly increasing delays.

A somewhat more complex example of the RBS principle is shown in Figure 1-2, wherein the schedule is nonuniform. In general, flights are scheduled with two minute interarrival times, but there are several periods during which no flights are scheduled. Thus, flights that are scheduled after an unscheduled period receive a better arrival time than they would have under a uniform schedule.

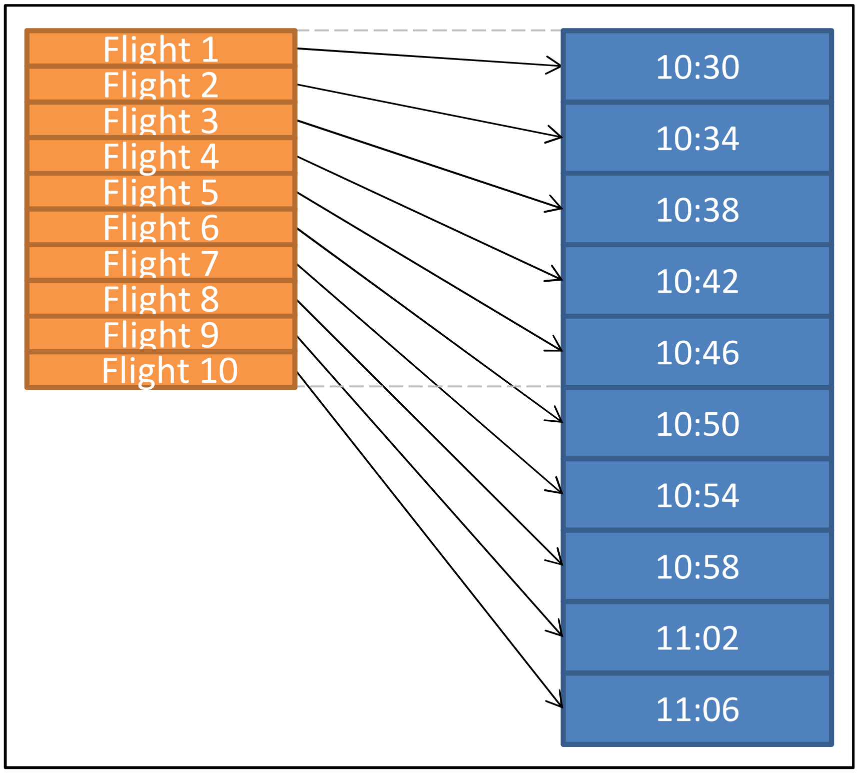


Figure 1-1 – Ration By Schedule with uniform schedule

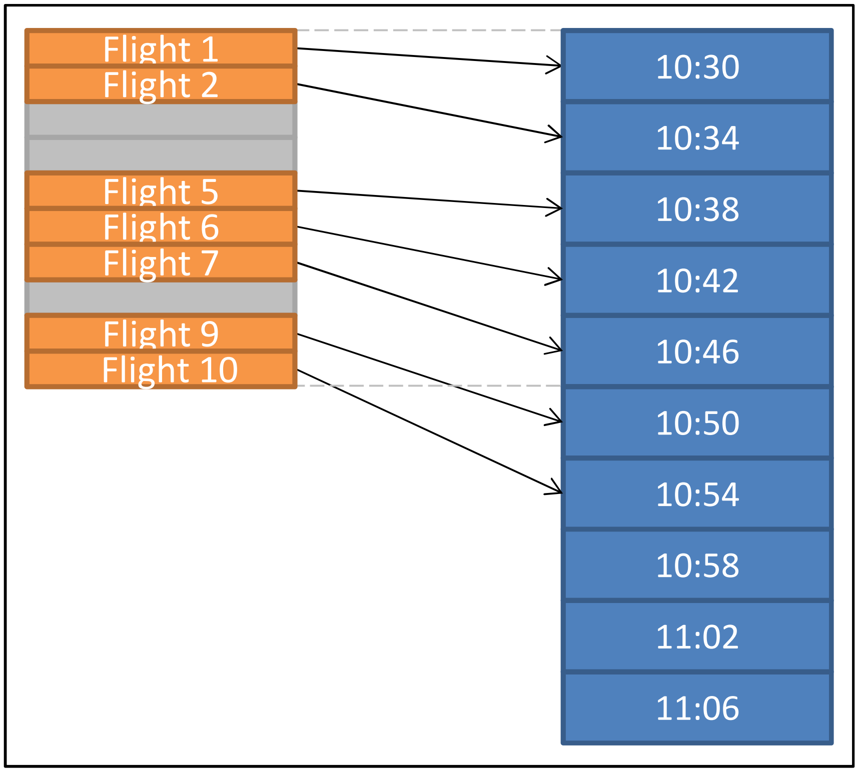


Figure 1-2 – Ration By Schedule with non-uniform schedule

Several alternate standards for rationing have been proposed to replace RBS. These have been developed in an effort to improve in some way upon the properties of the RBS allocation. These include rationing by aircraft size, by number of passengers (Manley 2008), by aircraft fuel burn, and by distance (Ball, Hoffman and Mukherjee 2009).

Of these, Ration By Distance seems to have the most interesting properties, in that it explicitly mitigates against uncertainty concerning the ending time of the disruption to minimize total expected delay. An example allocation for a limited set of flights is shown in Figure 1-3. The flights are assigned their best feasible slot in order of the length of the flight. This allows for the quick release of shorter flights if the disruption ends early. Clearly this comes at a severe cost to the equity principles established by RBS by heavily penalizing shorter flights. This property has been subsequently explored in (Glover and Ball 2010).

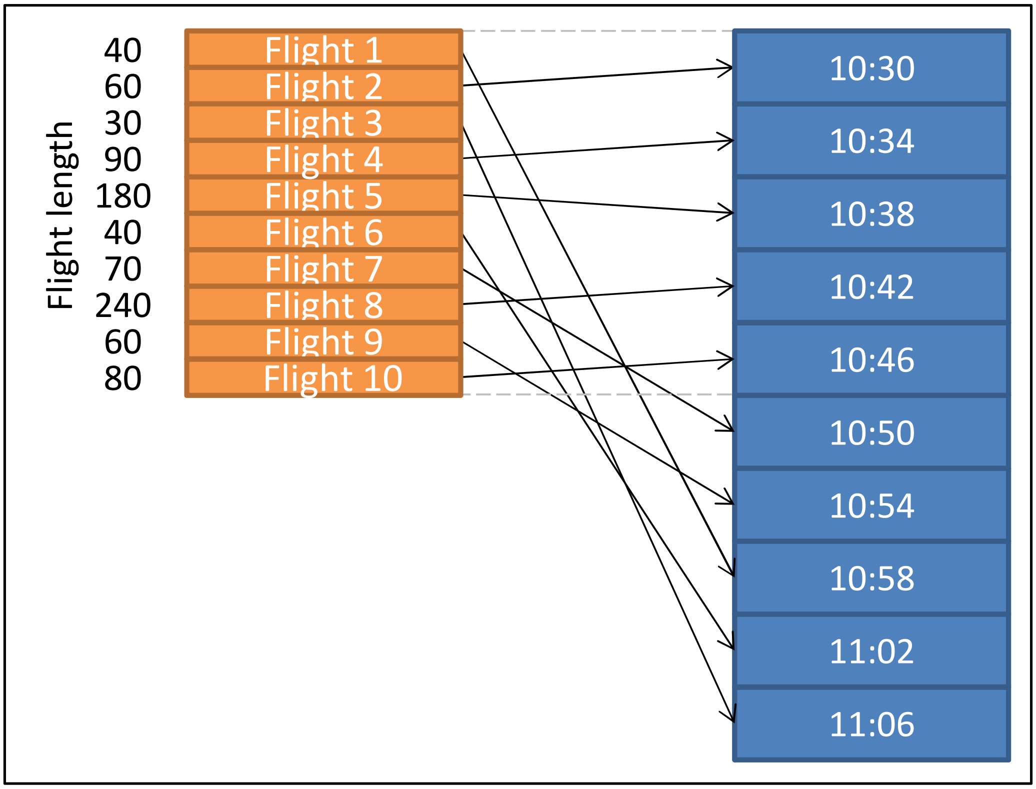


Figure 1-3 – Ration By Distance principle

In practice, the RBS procedure is more complex, as it incorporates exemptions and has several other algorithmic components that attempt to achieve greater efficiencies. However, at its core, the RBS allocation is formed simply by stretching the original schedule order of flights. While in the previous rationing examples, only a single airline was considered, in reality many airlines are included in the rationing process. The implications of this will be addressed, along with descriptions of the specific operational systems used to allocate capacity and variances from the above procedure, in the next section.

1.1.2 *Operational systems*

In the U.S., there are several tools that apply these rationing methods to demand-capacity imbalances in the U.S. Ground Delay Programs (GDP) and Airspace Flow Programs (AFP) handle this problem on the ground and in the air, respectively. These initiatives operate independently from one another and are employed sparingly – under nominal conditions they are typically not used.

In Europe, a more integrated approach is employed. The Central Flow Management Unit (CFMU) uses a dynamic heuristic to identify expected demand-capacity imbalances throughout each flight's route and constantly issue new control times to prevent them. Thus, the European system takes a much more control-oriented philosophy in contrast to the U.S. system, which puts a greater onus on the flight operator. This system is employed continuously, always adjusting flight control times.

Ground Delay Programs function according to the RBS principle outlined in the previous section. They have been in use in some fashion for a long time. They are planned by the FAA after it has identified a period of expected demand-capacity imbalance at some airport. Typically, they are planned several hours in advance and are expected to last for several hours. Only arrivals are explicitly controlled, as keeping flights on the ground, either departures to or from the airport under consideration, is inherently safer and less expensive than delays in the air.

An important feature of GDP's is that airlines have great flexibility to modify the flight-slot allocations assigned to them. Because each airline has their own internal business objectives, they may prefer to prioritize one flight over another, even to the point of cancelling a flight to move others much earlier. This may yield them tremendous benefit. They are only permitted to do this because they agree to share information about the cancellation in advance of the planned departure. This information sharing is a key part of the Collaborative Decision Making (CDM) paradigm in rationing capacity, as described in (Wambsganss 1997).

If the airline is unable to make use of all the capacity that it is allocated after it has made cancellations and substitutions, then other airlines flights are moved earlier in the allocation according to their schedule order. This process is known as compression.

Until the summer of 2006, airspace capacity was not explicitly rationed. At that time, however, to address increasing airspace congestion, airspace flow programs (AFP) were introduced, employing the same principles and software to manage disruptions as are used for GDP (Krozel, Jakobovits and Penny 2006). The development of the production system to implement AFP is described in (Brennan 2007). AFP provides a mechanism for airspace operators to directly control the flow of aircraft using a particular region of airspace. The regions of airspace controlled are drawn primarily from a set of predefined regions, as depicted in Figure 1-4.

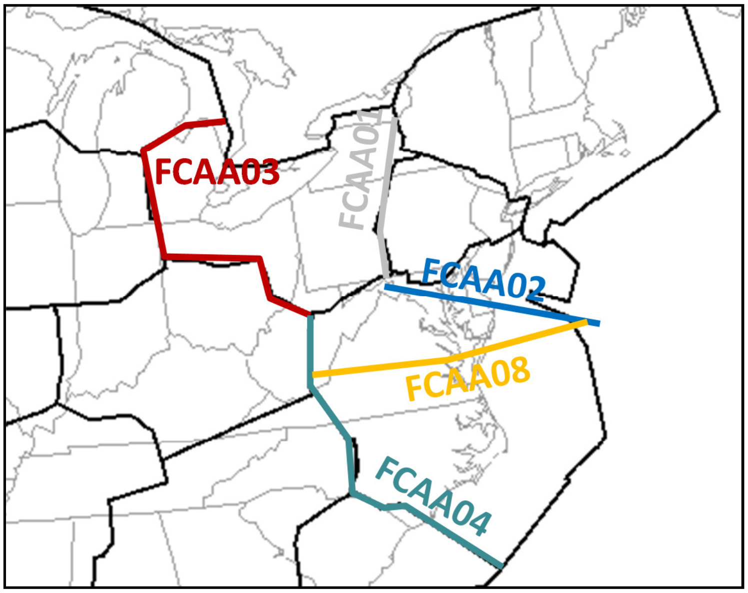


Figure 1-4 – Predefined AFP regions

The same software tools are used to implement GDP and AFP in practice, with the primary difference between any programs being the new interarrival times employed. In practice, however, the origin and veracity of these interarrival times is somewhat more challenging to address than for GDP. Airport runway capacity constraints driving a GDP are fairly straightforward to characterize because of the well-defined separation standards that must be enforced between operations. In the airspace, however, and particularly for a fairly large volume that might be employed in an AFP, the features defining the capacity constraints may be significantly more difficult to characterize. As a result, AFP capacity is estimated through expert opinion, and is not treated as a hard constraint. Violations of this notion of capacity are regularly admitted to help ensure feasible operational schedules.

It is important to note that the problem of coordinating capacity allocation for flights using multiple resources only appeared after the introduction of AFP. Prior to that time, only airport capacity was explicitly allocated. Because each flight may only arrive to a single airport, there existed no mechanism for conflicting allocations.

Another rationing method employed in airspace operations is the Ground Stop (GS). For whatever reason, the resource being considered is completely unavailable for some period of time. Causes of such stoppages include severe weather, equipment breakdown, and security situations, among others. No flights can be accepted, and so all that were scheduled to arrive during that time are delayed until after the stoppage. In a sense, GS is an extension of the above rationing principles, with a single period of extremely large interarrival times.

It is important to note that, in the U.S. system, each GDP or AFP functions independently of any others being used simultaneously – no account is made for coordination. As a result, flights may receive conflicting access times if they are using multiple congested resources. The inertia behind the U.S. approach dictates that this or some derivative system will continue to be used into the foreseeable future. However, increasing congestion at key points in the aviation system, both on the ground and in the airspace, requires that greater attention be paid to implementing some form of coordination between the slot allocations at various resources.

1.1.3 *Research and development*

The previous sections presented here explored the basic principles and the operational systems that are used in rationing capacity in the aviation system. Practical and cost considerations, as well as institutional barriers, keep such systems in use, even in the face of more complex systems that have the potential to streamline operations and reduce delays. A variety of models and systems have been developed from the research community to address these problems and to create comprehensive systems for allocating capacity. In this section, several of these models and systems will be described to provide the setting in which the research proposed here will fit.

Several groupings of models are considered here. Models may be divided simply by whether capacities, or any other data, are treated deterministically or stochastically. Alternatively, models may be divided according to the scope of resources and decision-making processes that they cover. These two divisions are illustrated in the Venn diagram shown in Figure 1-5. Some multi-resource models consider network effects, while others do not. In general, the least complex problems in this figure are the deterministic single resource models, while the most complex are those that consider multiple resources and network effects under stochastic capacity. The number of research efforts varies undertaken correlates well with these measures of complexity.

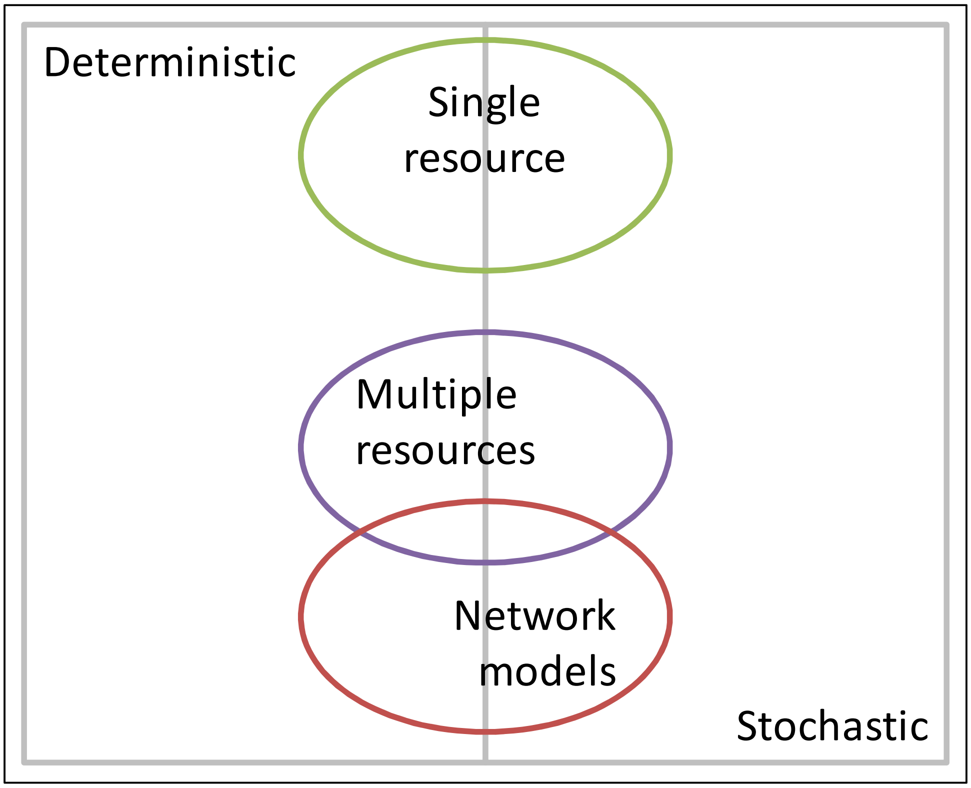


Figure 1-5 – Division of capacity allocation research

When the tremendous complexity of the air traffic system is considered in concert with the uncertainty associated with so many parts of the system, there clearly exists tremendous potential for the application of models to aid decision-making. However, most of the models proposed for these purposes do not explicitly adopt the same principles (e.g., RBS) as are used in practice, but take a more general delay or cost minimizing approach for some stakeholder group.

1.1.3.1 Single resource models

The first category of models considered are those that address capacity allocation at a single resource, either under deterministic or stochastic capacity assumptions. The first models designed for strategic air traffic management focused on allocating ground holding. The ground holding problem (GHP) was first systematically described in (Odoni 1987). This was formalized in (Terrab and Odoni 1993) to examine the Single Airport Ground Holding Problem (SAGHP). This linear optimization model minimized the total ground holding cost for a set of flights. An example of the scope of this problem is shown in Figure 1-6, wherein only flows into a single congested airport are the subject of explicit control.

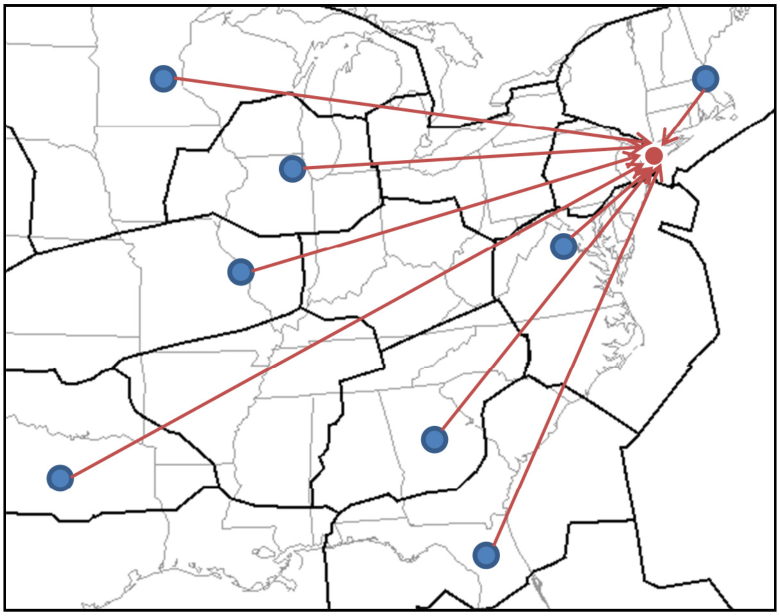


Figure 1-6 – Single airport ground holding problem scope

Several extensions to this deterministic SAGHP formulation were proposed, including (Hoffman 1997) and (Hoffman and Ball 2000), which extended this SAGHP formulation to include banking constraints requiring subsets of flights to arrive within small time windows. This addition models the connections that occur at hub airports to transfer passengers and cargo.

The other important dimension to consider in the SAGHP is the uncertainty associated with capacity values. In the previous papers, zero uncertainty was assumed. Clearly this is a limiting assumption, and several researchers have addressed it. A static stochastic integer program was proposed in (Richetta and Odoni 1993) to solve the SAGHP with uncertain capacities. Two later papers (Hoffman 1997) and (Ball, Hoffman and Odoni, et al. 2003) formulated a similar model to Richetta and Odoni, with the primary focus being on determining the optimal numbers of arrival slots to create under uncertainty. More recent efforts have also developed models that allow for dynamic updates, although their application is limited due to computational complexity (Mukherjee and Hansen 2007).

Efforts have been made to improve the computational properties of stochastic SAGHP models. The static stochastic SAHGP was shown to have strong computational properties under a limited set of conditions in (Kotnyek and Richetta 2006). In addition, (Glover and Ball 2010) introduced new stochastic SAGHP formulations that dramatically reduced solution times.

1.1.3.2 Multiple resource models

While early research focused assigning ground delays for a single resource with deterministic capacity, it quickly progressed to consider multiple resources simultaneously (Vranas, Bertsimas and Odoni 1994), (Vranas, Bertsimas and Odoni 1994) as the Multiple Airport Ground Holding Problem (MAGHP). This model focused on physical connections between aircraft operating multiple flights, although each flight was affected by at most one congested resource. This reflects a type of coordination in allocating capacity, however the scope differs from that examined here, and it did not represent realistic decision making ability. These early models considering multiple resources were computationally difficult. The scope of this model is depicted in Figure 1-7.

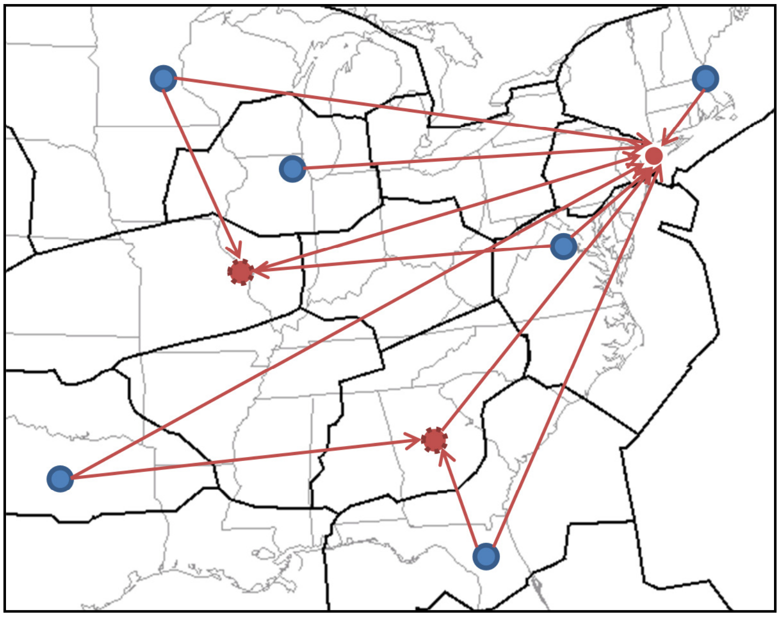


Figure 1-7 – Multiple airport ground holding problem scope

A significant body of research for ATFM models began with the introduction of the Bertsimas and Stock Patterson (BSP) model (Bertsimas and Stock Patterson 1998). This integer programming formulation, termed the Traffic Flow Management Problem (TFMP) attempted to capture the ground and airborne holding decision, along with airframe connectivity, in the same framework. It has reasonable mathematical properties that allow for it to be used for regional scenarios, but applying it to a nationwide problem is problematic computationally. This model did not directly address routing, but provides the base for several others that did. Evolutions of this model for multi-resource problems have addressed peculiarities of the European ATM system (Lulli and Odoni 2007), or have provided improved equity properties (Fearing, et al. 2009). Efforts have been undertaken to improve computational times (Rios and Ross 2008) (Rios and Lohn 2009) as well. Because of the model's inherent complexity, only limited efforts have been made thus far to consider stochasticity in capacity parameters (Chen 2009).

However, these Bertsimas models have seen limited use because they take a modeling approach incompatible with the operational patterns employed in the air traffic system. Primarily, they require knowledge of time-varying capacities for every resource, and plan accordingly. This lies in stark contrast to the operational approach of considering only congested resources for strategic intervention.

Finally, a separate group of optimization models for optimizing demand-capacity imbalances and for choosing flight routings have been developed in (Sherali, Staats and Trani 2003) and (Sherali, Staats and Trani 2006).

1.1.3.3 Network models

The third category of ATFM models examined here focus on allocating capacity in a network setting. Clearly there is significant overlap between these models and those highlighted in the previous section. The distinction is somewhat arbitrary, but lies primarily in whether the models include some notion of the network that connects the various resources in the airspace system. One of the earliest such network models for this problem was proposed in (Helme 1992), but this model was intractable because of its size. The general scope of these network models is depicted in Figure 1-8, wherein airports and airspace sectors are considered.

An essential difference in network models is whether they take an Eulerian perspective considering agglomerations of aircraft comprising flows, or a Lagrangian one considering many combinations of decisions for individual aircraft.

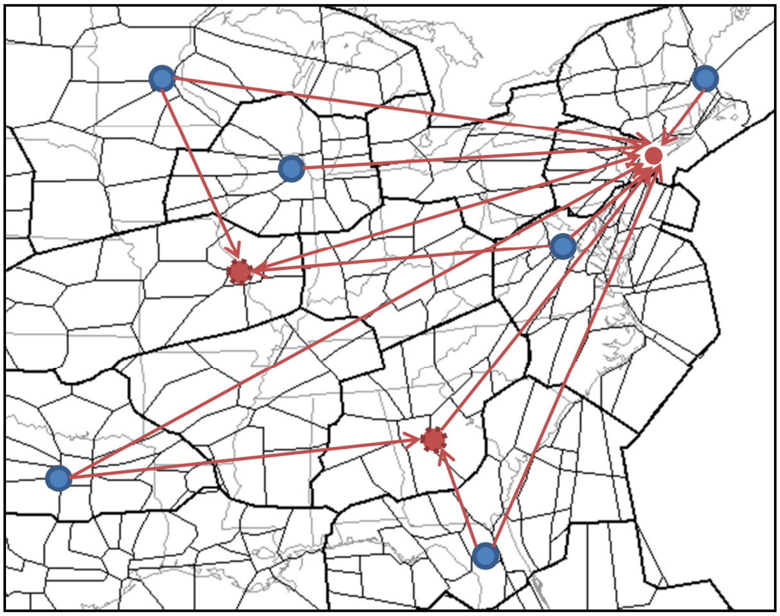


Figure 1-8 – Traffic flow management problem scope

Several significant research efforts have produced Eulerian models, including the original Helme model, building on the flow-based nature of the air traffic system. Such models typically retain superior computational properties to the individual flight models, but sacrifice tangible meaning. As a result, mechanisms must be developed to disaggregate the results of these models to utilize their results. Eulerian models have been developed with a control theoretic framework (Sridhar, et al. 2004), as well as a cell transmission framework (Sun and Bayen 2008). The NetFM model (Myers and Kierstead 2008) is another such aggregate model, but has improved computational performance. It models the NAS as a set of multicommodity flows, and is used to examine demand-capacity imbalance and tradeoffs between ground delays and alternate routing strategies.

The Bertsimas lineage of models takes a Lagrangian view more tangibly connected to the operation of the air traffic system. Several of these were described previously as multi-resource models, but they have some network properties as well. The same research team followed up the BSP model with a revised version that considered route choice in conjunction with the previous decisions (Bertsimas and Stock Patterson 2000). The additional dimension of complexity obviated some of the valuable mathematical properties of the earlier model, and as a result, this version has seen little use.

More recently, a reformulation of the first BSP model has been introduced to better include the routing decision (Bertsimas, Lulli and Odoni 2008) (BLO). This model seems to retain the valuable mathematical properties while incorporating the entire gamut of flight decisions: ground holding, airborne holding, rerouting. This model led to (Churchill, Lovell and Ball 2009), which represented a simplification of BLO to consider only congested resources. That model inspired the various formulations presented later in this dissertation.

An additional model, separate from those described above, was proposed in (Ganji, et al. 2009). This IP rations capacity under uncertainty for a single en route resource, but considers the additional dimension of allowing a flight to route around the disruption along the network present in the airspace.

The many research efforts presented in this section represent valuable contributions for ATFM. However, the single resource models do not provide the necessary complexity to model the problem coordination examined here. The multi-resource and network models focus on different relationships between resources, or on developing a comprehensive plan for all resources. Thus, the research presented here fills a gap in both physical and temporal scope in examining the problem of coordinating capacity allocation at several connected resources.

1.2 Coordination in ATFM

In operations in the United States, access to specific congested aviation resources is controlled by a system of capacity allocation wherein flights are assigned to slots at specific times. This U.S. approach is well-accepted and efficient, but it is not well-equipped to handle the problem faced when a single flight is included in more than one capacity allocation program, as may now happen when a flight plans to travel through congested airspace before arriving at a congested airport. The question of which, if any, initiative takes precedence over the others is not easily answered.

The number of flights affected by multiple resources is not negligible, as shown in Figure 1-9, using data from Metron Aviation Inc. Using data from summer 2008 for all days on which multiple resources were in use, this frequency chart depicts the fraction of flights visiting more than one. According to this sample, the mean fraction of flights affected by multiple initiatives is 11.1%. This data includes flights affected by GDP, AFP, and GS.

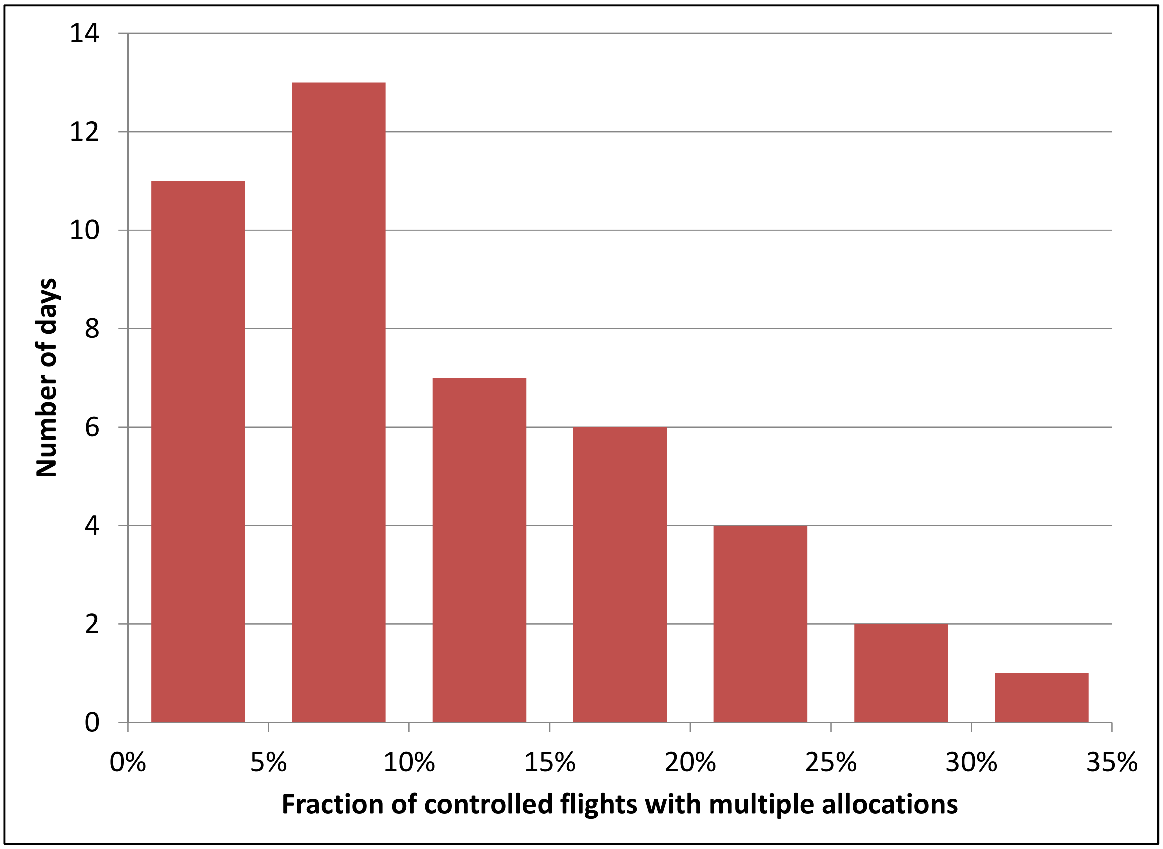


Figure 1-9 – Distribution of number of initiatives per flight

Operationally, these multiple capacity rationing initiatives operate independently of one another. The rationing procedures for each initiative use the same criterion – schedule, with some exemptions for flights already en route or coming from international destinations – but do not coordinate slot assignments between one another to minimize delay across multiple initiatives or multiple flight legs.

A key illustration of this failing is seen in the northeastern portion of the U.S., wherein a single flight may be affected simultaneously by initiatives to ration both airspace capacity to enter the region, as well as airport arrival capacity. This conflict is illustrated notionally in Figure 1-10 with flights classed into flows based on which resources they are using. Assume that two points of reduced capacity have been identified – a region of airspace (labeled FCAA03) commonly used for AFP controls, and an airport (JFK) commonly beset by congestion.

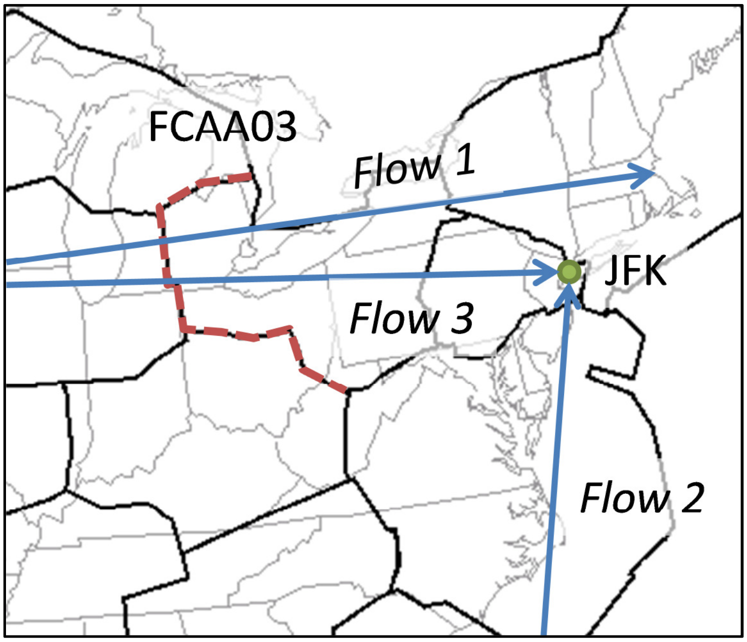


Figure 1-10 – Canonical case for coordination

Flow 1 comprises those flights passing through FCAA03 but not arriving at the compromised airport, while Flow 2 represents those flights arriving at JFK, but not using the congested airspace. In isolation, rationing the capacity of each of these resources is a well-solved problem, as Flows 1 and 2 can be treated separately without consequence. However, in this case, consider also Flow 3, which comprises those flights passing first through FCAA03 before arriving to JFK. It is these flights in Flow 3 that complicate the ATFM process in the U.S. paradigm, as they must participate in the rationing initiatives at both resources. Treated separately, there is no guarantee that the flights within Flow 3 will receive slot assignments compatible with one another at each resource.

To continue the example shown in Figure 1-10, assume that these two resources lie 90 minutes flying time apart, and that one of the flights in flow 3 received the slot assignments shown in the space-time diagram in Figure 1-11.

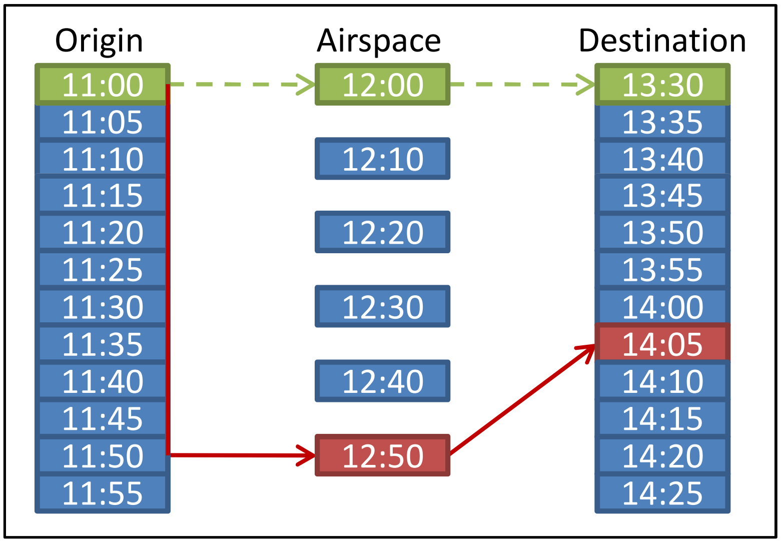


Figure 1-11 – Example slot allocation

This flight could wait on the ground at its origin airport until departing to use the 12:30 slot at FCAA03. However, to then meet its 14:05 slot at JFK, a 75 minute travel would be required. This likely represents an infeasible speed increase. Although the precise limit at which a speed increase or decrease becomes infeasible may vary with flight and operator, there clearly exists some bound both above and below. Further, this example considers only one flight. Obviously, there are many instances where this conflict would not exist, and many others in which it would be far more severe. In any case, choosing the flights to which an advantage should be granted, and those that should be disadvantaged is a challenging proposition.

A simple strategy to resolve this conflict is to prioritize one initiative or the other, while granting free passage at the non-prioritized resource at whatever time is most expedient. The assignment shown in Figure 1-12, wherein the airport initiative is prioritized and a new slot created in the airspace initiative, represents the system currently employed operationally by the FAA to resolve this incompatibility. Likewise, the airspace initiative could be prioritized, as shown in Figure 1-13. This example assumes simultaneity in planning, but the same phenomenon could certainly be observed when planning capacity allocations iteratively as well.

In either case, this approach causes several problems. This method of "creating" slots works to a certain degree in the airspace because the notion of airspace capacity itself is fairly ill-defined. For aircraft departing or arriving at an airport, the interoperation separation requirements are well-defined and flights generally operate close to those limits under congestion, essentially nullifying the ability to create slots in this fashion without sacrificing safety and violating protocol. In the airspace, however, considerably greater slack exists because aircraft do not operate so close to the separation limits. In addition, the very notion of what comprises an individual resource is somewhat ill-defined, as flights along several parallel, but far separated routes may be rationed together, when in fact it is not immediately clear that they are dependent operations.

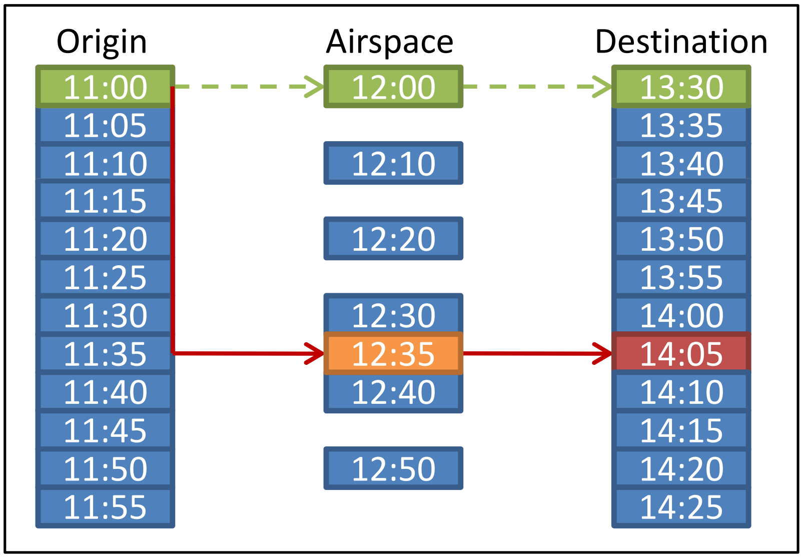


Figure 1-12 – Slot allocation for airport priority

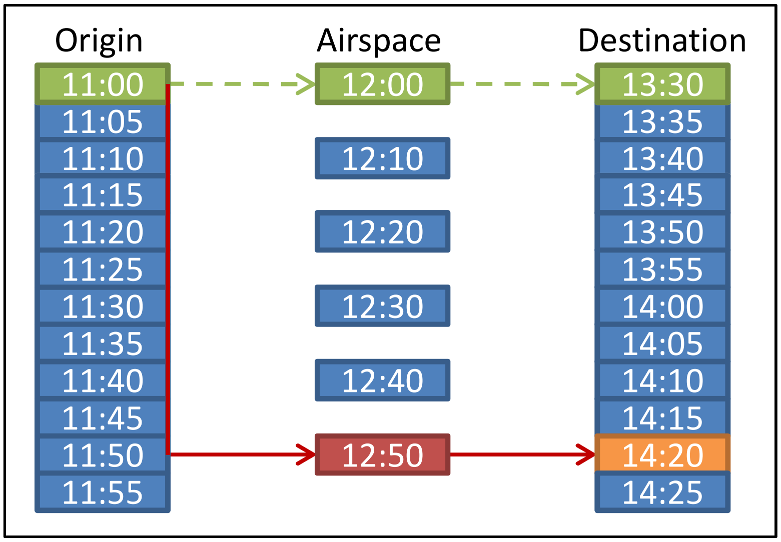


Figure 1-13 – Slot allocation for airspace priority

However, recognizing that better defining airspace capacity is an open problem, this dissertation will take the approach that, in the abstract, *some* capacity value (interoperation headways) can be defined for *some* airspace resource. Primarily, this allows for the development of models and for comprehensive analysis, but this is also employed because this is consistent with the approach long taken in managing air traffic at airports in the U.S.

Regardless of the capacity impacts, these approaches may create inequities for flights not able to receive such exemptions. It is this problem, in particular, that adds considerable complexity when considering many flights. The ultimate objective of a solution to such a problem is to balance the winners and losers (equity between users) while keeping overall delays at some reasonable level. The many possible combinations yielded by considering many flights create a complex combinatorial problem, and models presented in this dissertation address this problem from a variety of assumptions.

1.3 Contribution and contents

This dissertation makes several contributions, all related new approaches to finding solutions to the coordinated capacity allocation problem described in the previous section. This represents a new problem in both spatial and temporal scope, representing greater complexity than single resource problems, but less complexity than comprehensive ATFM models. Thus, neither the single resource models, nor the network models, including the Bertsimas lineage, are appropriate to address this problem.

- A new optimization model for coordinating slot allocations for flights using multiple congested resources is proposed and demonstrated. This model is novel in its approach to modeling the resources and their connectivity, as well as the unit-capacity construct employed. An analysis of the advantages of this approach is included. In addition, an analysis the objective of minimizing arrival delays, commonly used in ATFM, demonstrates a clear preference for single-resource flights in place of flights using more resources. This objective's analytic properties are used to develop a rule-based heuristic that produces quality solutions.

- Recognizing the biases induced by the two objective functions considered in the base model, new constraints and objectives are proposed that explicitly or implicitly regulate the equitable treatment of various users to enhance this optimization model.
- The impact of uncertainty on this coordination problem is explicitly quantified. A stochastic optimization model is proposed that explicitly includes these considerations to develop more robust slot allocations and recourse plans for each capacity scenario outcome.

The body of this dissertation is structured according to the three primary contributions, with each comprising an individual chapter in which models are formulated, analyzed, tested, and evaluated.

2. Deterministic coordinated airspace capacity rationing

This chapter examines the general problem of coordinating capacity rationing at multiple resources during air traffic flow management (ATFM) processes using several methodologies. Building on the example case of conflicting slot assignments shown in the previous section, the models presented in this section develop coordinated slot assignments for several flights using one or more of a series of congested airspace resources.

This chapter addresses this problem through several approaches. First, the general planning paradigm and modeling assumptions employed are outlined. Then, an optimization model for the problem is formulated. However, recognizing the difficulty in implementing such a system operationally, and for several other reasons to be discussed in detail later, a system of priority rules is described to mimic the optimal solutions generated by the optimization model. Finally, a case study comparing these various approaches is shown and several practical considerations are addressed.

The optimization model presented in this chapter is similar in spirit to the multi-resource models, in particular the Bertsimas family. However, as will be outlined, it focuses more specifically on a subset of the problems addressed by those models. It is more compact in some respects and follows several alternative planning paradigms.

2.1 Planning paradigm and assumptions

It is important to carefully outline the assumptions, particularly with respect to scope and authority, built into modeling this problem. The first, and perhaps most impactful, assumption is that resource capacity is treated deterministically. Explicitly, the time-varying evolution of capacity at each resource is assumed to be known with certainty at the time of planning. This is a strong assumption, but valid for several reasons. First, developing a deterministic model is a precursor to any useful stochastic model. Second, it is largely consistent with the type of data available for use for ATFM. Finally, it is consistent with operational practice.

Of critical importance is the scope of airspace resources considered in modeling this capacity allocation problem. Only resources expected to be congested are considered. Flights that do not use a congested resource are not considered. Thus, some rudimentary predictive capabilities must be employed to remove from consideration those resources for which congestion is highly unlikely. This must be done using some other model, expert judgment, or a combination of the two. While this assumption potentially adds a bias based upon the selection of resources, it is justified in several ways. First, some regions of the United States, particularly the Upper Midwest, will naturally have a very low traffic density. Undue control of such resources is an inefficient use of resources. Further, this approach is consistent with the philosophy employed in the U.S. air traffic system, wherein control is only exerted when it is explicitly needed. This lies partially in contrast to the approach employed in other parts of the world. Finally, excluding uncongested resources allows for simpler and more-compact models than considering all resources.

Another assumption employed in addressing this problem is that, in general, plans for all resources being considered are developed simultaneously. This stands in contrast to the piecemeal approach employed in practice, which is largely based upon the lead times and availability for the capacity data used. Strategies for relaxing this assumption will be discussed for each methodology employed.

Finally, the models presented do represent, in some respects, a greater degree of control than is currently exerted by system operators. The paradigm is compatible, nevertheless, with the principles of collaborative decision making (CDM) that have been so widely adopted in ATFM. These CDM principles (Wambsganss 1997) encourage information sharing between users and system operators to enhance planning and increase both equitable and efficient outcomes. The decisions developed in this model represent the initial assignments that would be made for flights, but there is no reason that individual airlines or users, with their collections of slots at each resource, could not perform their own swaps or trades to meet internal objectives. While trading slots may detract from the system-level objectives espoused by this model, it represents the ability of users to optimize their operations within the construct provided.

2.2 Optimization formulation

The first approach presented in this chapter for coordinating ATFM decisions is a linear optimization model. The broad objective of the proposed model is to ration access rights to each of several capacitated airspace resources, as depicted notionally in Figure 1-10, while minimizing delays. Specifically, the model assigns flights to arrival times at each of a sequence of congested resources that the flight encounters between origin and destination. A resource may be an airport, some congested portion of airspace, or any other airspace resource of finite capacity. The model takes as input a list of these resources and their associated time-varying capacities, as well as a list of flights and their respective scheduled times to arrive at each resource. The outputs of the model are the slot times to which each individual flight is assigned.

Structurally, allocation at each resource is considered as an assignment problem, but side constraints are added that link each of the resources together and guarantee that each flight using multiple resources receives compatible slot assignments. Although the structure of the model is different, this concept was proposed in (Churchill, Lovell and Ball 2009), wherein only those regions under adverse conditions are expressly controlled. However, the application considered here is more specific than the system-wide plan developed in that work.

2.2.1 *Input data*

Several input data are required for this optimization model, categorized generally as pertaining to individual flights or to capacitated resources. Each will be described in detail in this section.

2.2.1.1 Flight data

Both schedule and path data are required for each flight considered in this formulation. If flights are indexed as *f* in the set *F*, and resources as *i* in the set *I*, then the scheduled (or planned) arrival time of flight *f* to resource *i* is defined as α_f^i. Flight departure time is not required for this formulation, but is defined as δ_f. All flights are treated as having identical characteristics.

This model assumes that routes for each flight are fixed. Flight paths are defined using two data constructs: V_f defines the set of capacitated resources a flight *f* visits, and N_f^i defines the next resource a flight *f* travels to after resource *i* to maintain ordering. For a flight to be included $\left|V_f\right|$ must be greater than zero. Example flight paths are shown for two flights (*f*, *g*) in Figure 2-1. In this example, flight *f* visits resources *i*, *j*, and *k*, and so $V_f = \{i, j, k\}$. Flight *g* visits resources *l* and *j*, and so $V_g = \{j, l\}$. The data for the next resources are shown in the figure.

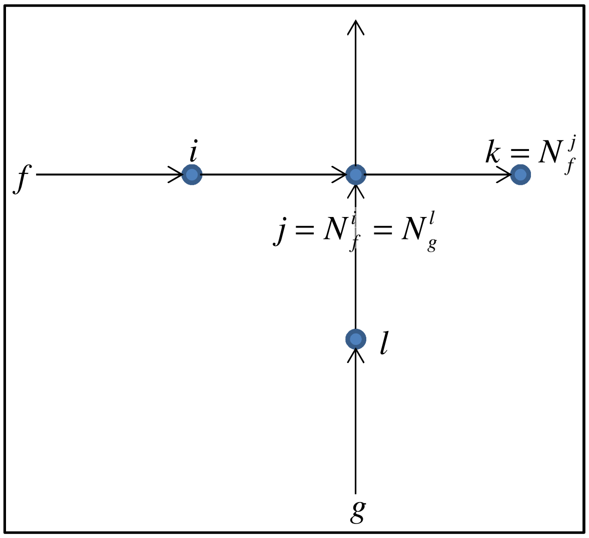


Figure 2-1 – Flight path definitions

2.2.1.2 Resource data

Resources for which capacity is to be rationed are indexed as *i* in the set *I*, and may include airports or critical airspace regions. Each initiative *i* has its own independent slot set S^i, with each slot *s* in S^i having a time marker τ_s^i. The number of slots must equal or exceed the number of flights, as shown in (2.1).

$$|S| \geq |F| \tag{2.1}$$

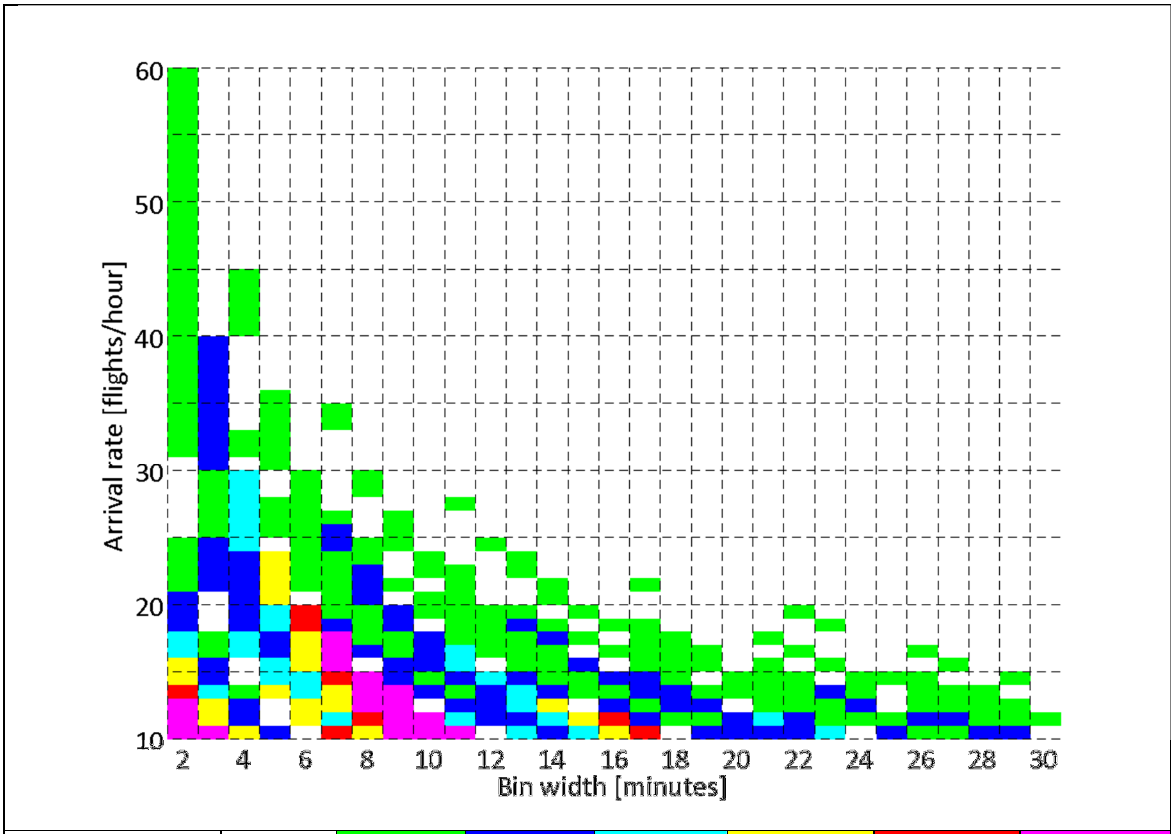
This condition is trivial to enforce when developing an instance. Many additional slots with small headways may be created after the planned rationing time to represent the unconstrained operation of the airport. This simulates the reality at most airports, at which permissions to operate revert to a first-come, first-served system after rationing has concluded. Further, absent this condition, the problem will lack a feasible solution.

Resource capacities are obviously critical in this model. There are two important points to address concerning the capacities employed. The first of these focuses on the actual values, while the second concerns the description of these capacities in the model itself.

In practice, as described in §1, flights using multiple resources are granted an exemption through airspace resources, effectively prioritizing the allocations made at airports. This practice is based upon the notion that capacity in the air is a soft constraint. While there is some truth to support this, some maximum capacity for any arbitrary region also exists based upon geometry, controller workload, and a variety of other factors. Modeling these as such is beyond the scope of this research. However, clearly there must be some limit to the number of exemptions that may be feasibly granted, and the research described here assumes that such a value has been identified. The actual value used in the model may be somewhat less than this upper bound to provide additional margins of safety, but in the base model and the results presented, the capacity values represent hard constraints. Further, capacity at each resource is treated independently of the anticipated fleet mix. All flights are considered as equivalent with respect to the "amount" of capacity that they utilize.

The descriptions of capacities employed in this formulation represent a departure from previous work and warrant additional exploration. Many previous ATFM models, including (Bertsimas, Lulli and Odoni 2008) and (Ganji, et al. 2009), among others, consider capacitated time periods. Under this construct, the entire planning period is partitioned into a lattice of time blocks of equal length. Each time block is assigned some capacity, typically greater than one.

For example, if an arrival rate of 30 flights per hour is to be modeled using 15-minute time blocks, then in 1 hour, the four time blocks in each must have values of 7, 8, 7, and 8 in some order. Although the intention is to model constant headways of 2 minutes, headways will alternate between 1.88 and 2.14 minutes because of the varying rates. Thus, for specific combinations of time block length and capacity, a sawtooth-type pattern of capacity values is developed. The errors in modeling headways induced by this sawtooth pattern are shown in Figure 2-2. The various shades in this figure indicate the degree of severity of the error. In general, the errors in modeling headways induced by time blocks are relatively small for reasonable combinations, but are nonzero.

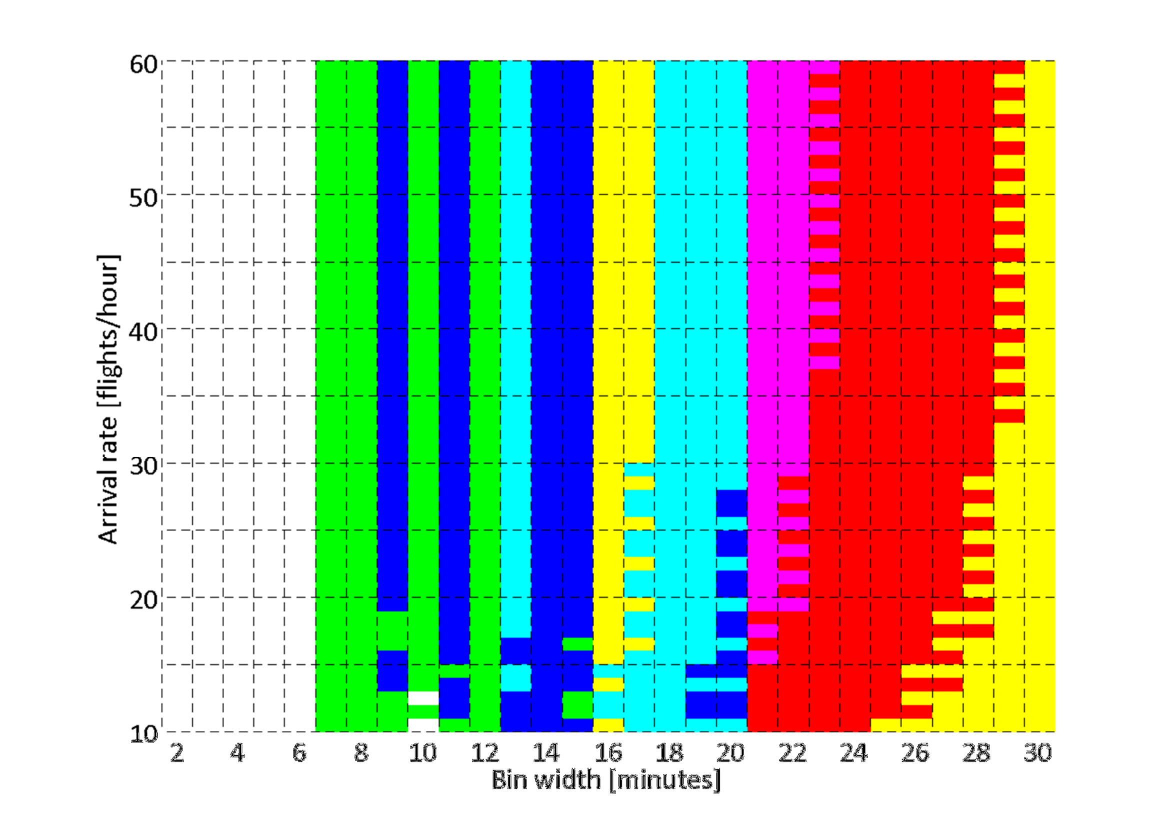


Error (seconds per flight)	0–30	30–60	60–90	90–120	120–150	150–180	180+

Figure 2-2 – Error in headways induced by time blocks

In contrast, capacities in this model are represented by time slots of unit capacity. This provides several advantages over the previous construct, beginning with the ability to model constant headways without the sawtooth phenomenon. In addition, greater flexibility is provided for dynamic situations, wherein the headways between every subsequent pair of slots may be defined individually, as the model is not tied to fixed time blocks. To accomplish this, slot indices are decoupled from the time associated with that slot. This adds an additional qualification (discussed later) to some of the summation terms. Thus, resource capacity is specified as a list of slots with individual times, rather than a list of time blocks each with associated capacities.

In addition, unit capacity slots provide the advantage of precisely specifying estimated flight arrival times within the model. When using time blocks, the model is forced to assign each flight within that block the same nominal arrival time for the purposes of determining delays. In truth, however, those flights' actual arrival times must be spaced somewhat uniformly throughout that time block, as simultaneous arrivals would be both inefficient and infeasible. Thus, this approximation induces an error in measuring assigned delays. This approximation error is compounded by the sawtooth pattern needed to model arrival rates that are not multiples of the time block length. The impact of these effects is depicted in Figure 2-3, with the color scale indicating the average delay induced by this approximation effect.



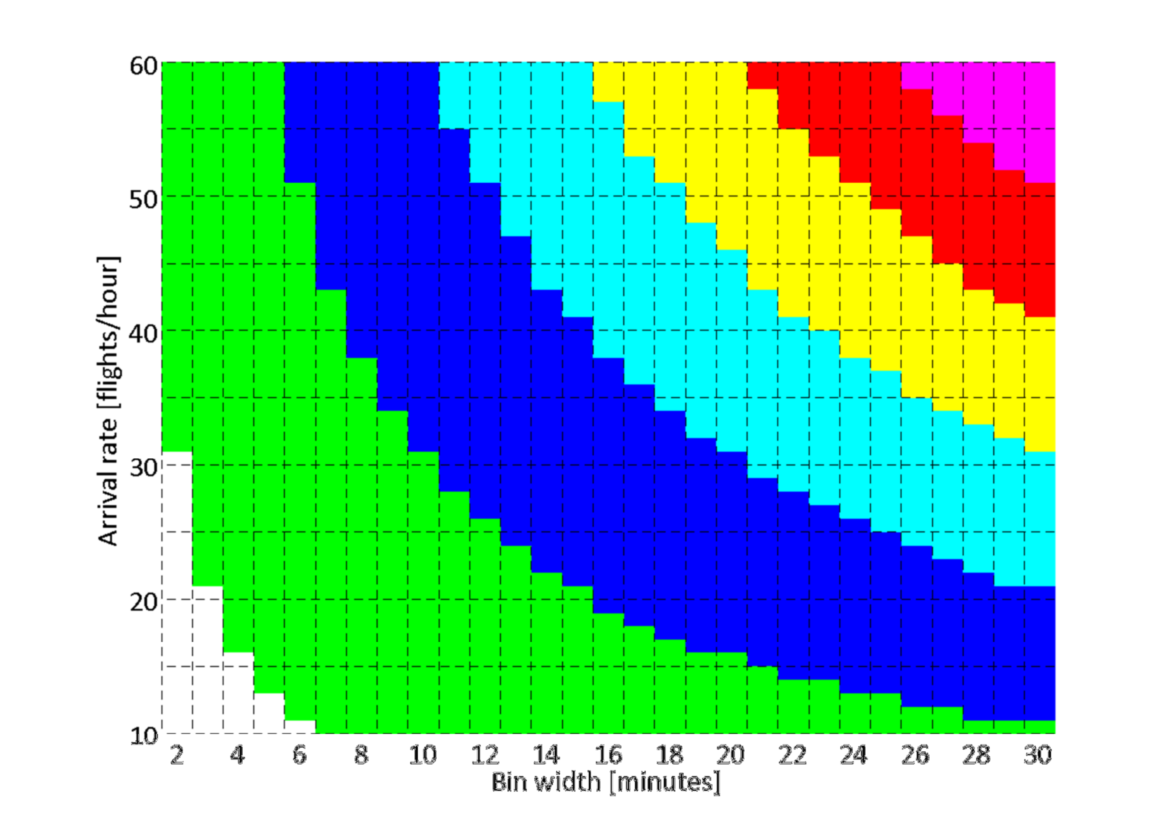
Error (minutes per flight)	0–3	3–6	6–9	9–12	12–15	15–18	18+

Figure 2-3 – Error in delay measurement induced by time blocks

The above two analyses represent the essential tradeoff required in using time blocks for modeling such problems: decreased bin width reduces error induced by arrival times, but this comes at the cost of increased headway approximation error. This trade is avoided in developing the formulation presented in this section through the use of unit capacity slots. Additionally, although the formulation affords these increases in precision, it remains compatible with the time block methodology. The reverse is typically not true of other such models.

Whereas the use of slots instead of time blocks avoids the approximations mentioned above, it comes at the cost of increased formulation size. Figure 2-4 shows the ratio of the number of unit slots required to the number of time blocks required to model a variety of capacity rates and bin widths. For a given bin width, the number of time blocks required is 60 times its inverse, while the number of slots required is simply equal to the arrival rate. As can be seen in this figure, the region over which the slot system reduces formulation size is small and relatively unimportant. However, the region over which the ratio is below a factor of five is fairly large and covers a number of useful and realistic points. That said, the time block model is clearly superior in terms of reduced formulation size.

The large increase in the number of entities needed to represent capacities does not translate directly to the same increases in formulation size. Conditions on feasible arrival times for each flight at each resource are used to limit the number of decision variables created, helping to control this increase in complexity.



Error (ratio of number of entities)	0–1	1–5	5–10	10–15	15–20	20–25	25–30

Figure 2-4 – Difference in number of variables required

Although the formulation presented in this chapter considers only deterministic capacity, this description of capacity provides advantages when considering a stochastic formulation. In that case, this methodology may greatly simplify formulations and solutions. For example, in cases in which capacity may only increase over time in various scenarios, this paradigm greatly simplifies solutions, as flights will stay assigned to the same slot index, but the operation time associated with that index will decrease.

Weighing the tradeoff between headway errors, delay measurement errors, formulation size, and flexibility for future model developments, unit capacity slots will be used in this formulation.

2.2.2 *Decision variables*

The decision variables x_{fs}^{i} are constrained as shown in (2.2), taking a value of one when flight *f* is assigned to slot *s* at resource *i* and zero in all other cases.

$$x_{fs}^{i} \quad binary \qquad\qquad \forall f \in F, i \in V_f, s \in Q_f^i \qquad (2.2)$$

Generically, the set of feasible slots for each flight is identified as Q_f^i. The simplest definition for this set is shown in (2.3), in which any slot after the flight's scheduled arrival time becomes feasible.

$$Q_f^i = \left\{ s \in S^i : \tau_s^i \geq \alpha_f^i \right\} \qquad (2.3)$$

The inclusion of the $\tau_s^i \geq \alpha_f^i$ condition helps to reduce the size of the constraint matrix by eliminating unnecessary decision variables, as was addressed previously. In principle, the excluded variables could be considered, but would be necessarily fixed to zero, because, by policy, a flight cannot be assigned an arrival time before that in its published schedule.

2.2.3 *Assignment constraints*

The first constraint set is shown in (2.4). This enforces the condition that each flight must be assigned to exactly one slot in each rationing initiative that it visits. This constraint is visualized in Figure 2-5 – exactly one of the highlighted arcs will be selected.

$$\sum_{s \in Q_f^i} x_{fs}^{i} = 1 \qquad\qquad \forall f \in F, i \in V_f \qquad (2.4)$$

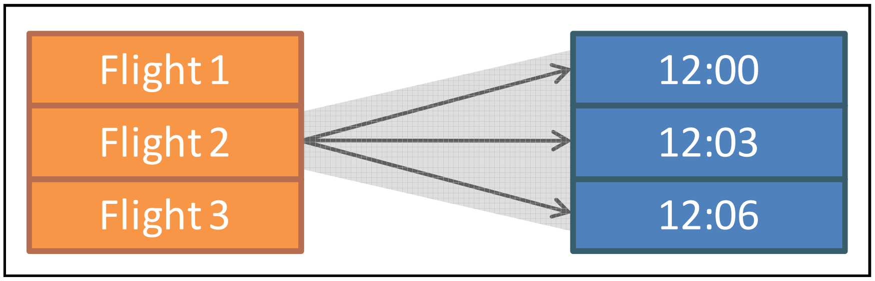


Figure 2-5 – Assignment constraint I

The second constraint set, (2.5), enforces the capacity of each slot at each initiative to be at most one flight. As discussed, the construct of using single-flight slots is somewhat unique. The structure of this constraint is shown in (2.5), wherein at most one of the highlighted arcs will be selected. This constraint could be modified to match precisely with the assignment problem if it were changed to equality, so long as a number of slack flights with zero cost were created to fill the extra slots. This adds complexity to the formulation, however, and provides little tangible benefit.

$$\sum_{\substack{f \in F: \\ i \in V_f}} x_{fs}^{i} \leq 1 \qquad \forall i \in I, s \in S^{i} \qquad (2.5)$$

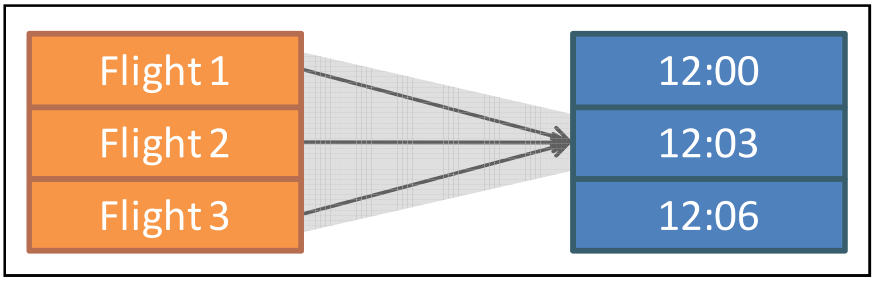


Figure 2-6 – Assignment constraint II

Although this model is described in the context of unit capacity slots, the formulation is compatible with time blocks allowing multiple flights as well, so long as the right-hand side of (2.5) is changed to reflect the increased capacity of each time block. The reverse is not true for other ATFM models, as the constraints linking multiple resources rely on the uniformity of the underlying time blocks and tend to include them in indices themselves, rather than as conditions on summations as here.

2.2.4 *Linking constraints*

The two constraint sets shown above, in isolation, will allocate flights to slots at only a single resource. The objective of this research, however, is to create feasible slot allocations for each flight using multiple resources. As a result, linking constraints are added to these assignment problems. A notional depiction of the structure of these linking constraints is shown in Figure 2-7.

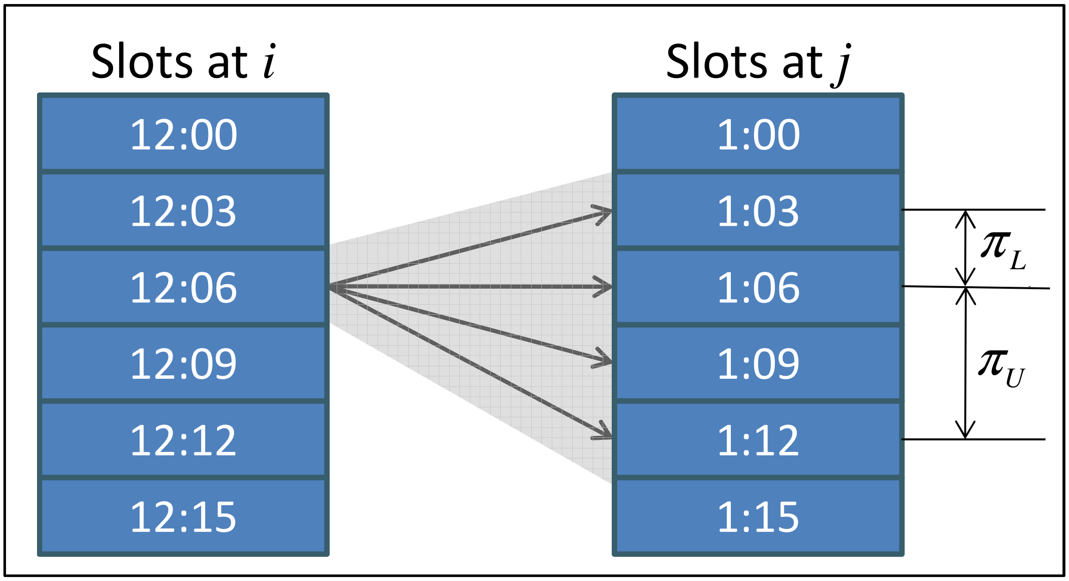


Figure 2-7 – Feasible range example

In this example, two resources are located 1 hour apart; thus, if a flight is assigned the 12:06 slot at the first resource, then the 1:06 slot at the second resource would be preferred. However, to help ensure feasible solutions, some slack is provided around that ideal assignment. In this example, this flight may be assigned a slot three minutes earlier, or up to six minutes later. The physical reality being modeled is that a flight can only increase speed so much within its own performance limits, and within the limits imposed on it by the air traffic system. Likewise, a flight may only slow down so much or enter an airborne holding pattern for so long, based largely upon its fuel load. It is likely, however, that this time limit is longer than that for increasing speed, as is reflected in the figure. These values are parameters of the model. Mathematically, the constraint set that links together these multiple resources is shown in (2.6)

$$x_{fs}^{i} - \sum_{k \in R_{fs}^{ij}} x_{fk}^{j} \leq 0 \qquad \forall f \in F, i \in V_f, j = N_f^i, s \in Q_f^i : \left| N_f^i \right| > 0 \quad (2.6)$$

If x_{fs}^{i} takes on a value of one, indicating that flight *f* is using slot *s* at initiative *i*, then one of the feasible slots in the following initiative must also take on a value of one. Because of the assignment constraints however, only one of these subsequent slots may be selected. In the case that x_{fs}^{i} is zero, then it is still possible for one of the feasible slots in the subsequent initiative to be used because the feasible ranges in *j* for different slots in *i* may overlap. If only unconnected initiatives are considered, then the value N_f^i is always empty, and none of this constraint set is present in the formulation. In that case, the formulation becomes separable for each initiative.

The range R_{fs}^{ij} is defined in (2.7) to control the feasible reassignment range for downstream resources. It defines the time period during which flight *f* could feasibly arrive at initiative *j*, conditioned upon its using slot *s* in initiative *i*. It begins at the sum of the slot time at the upstream resource (τ_s^i), the inter-resource travel time ($\alpha_f^j - \alpha_f^i$), and the negative of the early arrival parameter π_L. It ends at the sum of the earlier slot time, the inter-resource travel time, and the late arrival parameter.

$$R_{fs}^{ij} = \left\{ k \in S^j : \max\left(\alpha_f^j, \tau_s^i + \alpha_f^j - \alpha_f^i - \pi_L \right) \leq \tau_k^j \leq \tau_s^i + \alpha_f^j - \alpha_f^i + \pi_U \right\} \quad (2.7)$$

The implicit assumption in this model is that the values of the early arrival parameter π_L and the late arrival parameter π_U will be small. They are not intended to permit the assignment of larger airborne delays to develop feasible slot assignments – only to allow a small amount of slack to accommodate inconsistent lattices between resources. The early arrival parameter would likely be smaller than the late arrival parameter, simply because the ability of an aircraft at cruise speed to decrease speed is greater than its ability to increase speed.

The magnitudes of these parameters directly affect the existence of feasible solutions to this problem. As each tends to zero, the likelihood that a feasible solution exists decreases, as each flight using multiple resources has very few options for subsequent slots of which to make use. In practice, it would be difficult to reach this limit simply because there would not be an overwhelming number of flights with this limited flexibility. However, in a pathological case, it may be necessary to increase the magnitude of each of these parameters to generate a feasible solution.

2.2.5 *Objective functions*

Most useful objective functions for this problem will attempt to minimize some function of flight delays. Conversely, they may be viewed as maximizing some metric of system efficiency. This is consistent with virtually all other ATFM models. Delay is often a focus because of its direct impact on measurable system performance, as well as its direct cost to aircraft operators.

However, in this formulation, careful attention will be paid to precisely which delays are being minimized. Delays may be aggregated by flight or by resource, as listed below:

1. All delays at all resources: some delays are double counted, as a subset of flights uses multiple resources.
2. All delays at specific resources: may result in double counting, and may favor certain resources.
3. All flights upon arrival: reflects the delays truly experienced by flight operators and passengers. This is a special case of item 2, with each flight's destination being the specific resource for that flight.
4. Subset of flights upon arrival: favors some flights.

Several additional considerations are included in designing this objective function. Because this model treats capacity deterministically and sufficient capacity at the origin airport is assumed, all delays are planned to be taken on the ground before departure. Thus, there is no need to consider a cost differential between air and ground delays. A comprehensive model that schedules all points along a flight’s route, or one that considers capacity stochastically, however, fails this assumption. The early and late arrival parameters are assumed to induce delays sufficiently small as not to warrant inclusion beyond the base delays for the individual flights.

In addition, a superlinear function of delay length is used to favor the assignment of two short delays over a single long delay. This principle contributes to equity between different flight operators because flights that are similar a priori are assigned similar delays, as has been employed in previous ATFM models including (Bertsimas, Lulli and Odoni 2008) and (Churchill, Lovell and Ball 2009).

Two objective functions, items 1 and 3 in the above list, are considered in greater detail. The first objective, referred to as the “total delay objective,” represents the sums of all “delays” assigned at each resource, as defined in (2.8).

$$\min z = \sum_{f \in F} \sum_{i \in V_f} \sum_{s \in Q_f^i} \left(\tau_s^i - \alpha_f^i \right)^{1+\varepsilon} x_{fs}^i \qquad (2.8)$$

In this case, the measurement of delay is always taken with respect to the planned arrival time at each resource in a flight’s path, rather than singularly at the flight’s destination. Thus, some delays may be considered as “double counted” according to (2.8) because they are counted twice but truly impact the flight only once, upon its arrival at its destination.

However, this objective is desirable to consider for several reasons. First, it is consistent with operational practice, in that delays at each and every capacity rationing initiative are included. Obviously there are interventions to slot assignment made through the use of the model, but the total delay objective nonetheless calculates the delay metric identically. At a deeper level, this objective seems as if it should encourage equitable treatment of flights because the mass contributed to the objective function from each flight is comparable to the complexity it induces in the system. The implications of this will be explored in the next section.

The second objective, considering only the delay at the flight's final initiative, is shown in (2.9). It will be referred to as the "final delay objective." These final resources are identified through the use of the condition stating that a flight has no subsequent resource to visit ($\left|N_f^i\right| = 0$) for the flight to be included in the summation.

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| = 0}} \sum_{s \in Q_f^i} \left(\tau_s^i - \alpha_f^i\right)^{1+\varepsilon} x_{fs}^i \tag{2.9}$$

In the case that a flight's last controlled resource is not its destination airport, the model assumes that the delay assigned at that resource translates directly to the flight's destination. This objective is considered for several reasons. The first reason is that it is a very popular metric for other ATFM models and warrants consideration in this formulation for that reason alone. Again, however, the reasons for consideration are more complex. This objective function represents a more systematic view of the inherent network structure of the airspace system, in that it should automatically include interactions.

These two objectives seem to pose divergent views of the ATFM planning process. The first represents, to a degree, the independence of each resource included in the current system. The second caters to a more-robust, system-level objective by minimizing the delays that are actually experienced by each flight. The implications of these differences are quite significant, and will be explored analytically in the next section.

2.2.6 *Objective function implications*

The implications of the two objective functions outlined above are not obvious upon first analysis. To summarize, the final delay objective tends to prioritize flights using fewer resources, while the total delay objective tends to maintain schedule order. Although the final delay objective is apparently more desirable from a systematic point of view, this bias has serious implications for its utility. Conversely, the total delay objective seems somewhat obtuse, but the principles underlying it, of maintaining schedule order, are extremely valuable and consistent with accepted equity practices. Each of these points will be explored analytically in this section.

Before proving the analytic properties of each objective function, it is valuable to explore the notion of fairness implicit in the final delay objective a bit further. There is a reasonable argument to be made that prioritizing flights by the number of congested resources used is a valid rationing system. This follows from several arguments. First, flights that use multiple resources introduce complexity into the airspace system and should have to bear that cost, rather than distribute it to other users.

Further, some congested resources in the en route airspace may be avoidable by flights making use of alternate routings. It might be argued, for example, that convective weather patterns appear more often in the summer in some areas than others, and a flight operator may be able to choose avoiding the potentially risky portion of airspace, even if this extends the trip distance somewhat. Therefore, flight operators that choose to use these resources should be forced to internalize this cost. This argument faces one significant counterpoint – namely, identifying which resources might be avoided by which flights is a highly loaded question. Clearly airports are not avoidable resources, despite the fact that flight operators chose to allocate some of their resources to that airport. It is likely that any user would argue that making use of any "optional" en route resource was strictly necessary, when in fact there may be considerable controversy. Adopting an arbitrary paradigm of this nature clearly introduces a considerable number of complications.

Therefore, in this research, no position is taken as to advocating for either the total or final delay objective functions over the other. Both are explored and presented with respect to their individual strengths and weaknesses.

2.2.6.1 Identical schedule case

To demonstrate analytically the properties of these two objective functions, a case of three flights (1, 2, and 3) using two resources (*i*, *j*) will be examined. Flight 1 uses both resources, while Flights 2 and 3 each use one of the resources. Assume that Flights 1 and 2 have the same scheduled time to resource *i* and that Flights 1 and 3 have the same scheduled time to resource *j*. This schedule is depicted notionally in Figure 2-8, with the arrows and numbers representing flights and the circles representing slots in each program.

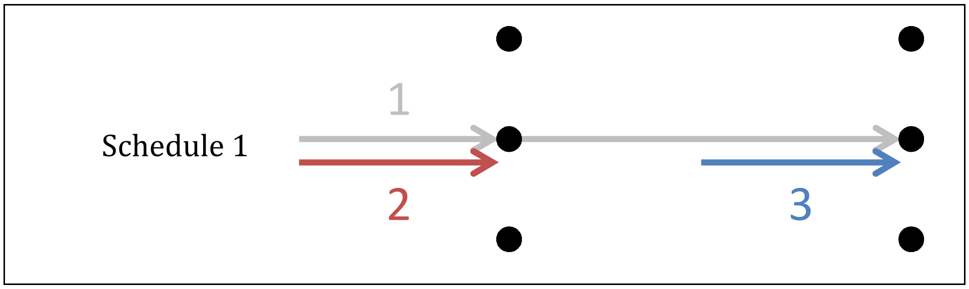


Figure 2-8 – Identical schedule

There are two possible allocations, identified here as A and B, for these three flights, depicted in Figure 2-9. The difference between these two allocations lies in whether the flight using both resources is given the first pair of slots, or if the other two flights are awarded those earlier slots. An explicit assumption is that each of these allocations is feasible. This means that the slots to which these flights are assigned are associated with sufficient delay so that the earliest slot has a time equal to or greater than the maximum of the flights' scheduled times.

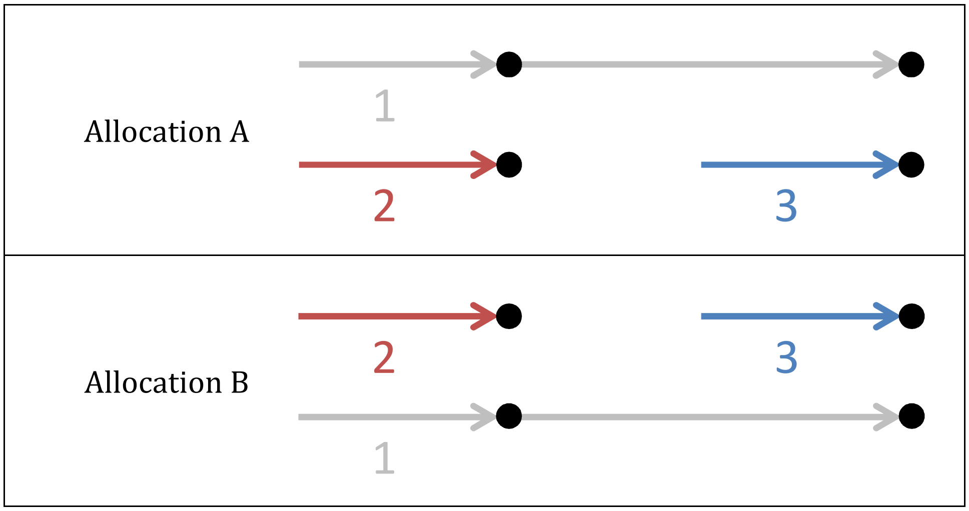


Figure 2-9 – Feasible allocations

Now, both allocations are compared according to the objective functions outlined above. The value d_f indicates the delay assigned to flight f under Allocation A. Table 2-1 shows the contributions to the total delay objective function, using the superlinear function of delay length, from each allocation, as well as the total contribution from the three flights under consideration. It is clear that the model should be indifferent regarding choosing one of these two allocations as part of the optimal solution when using the total delay objective function.

	Allocation A	**Allocation B**
Flight 1	$2d_1^{1+\varepsilon}$	$d_2^{1+\varepsilon} + d_3^{1+\varepsilon}$
Flight 2	$d_2^{1+\varepsilon}$	$d_1^{1+\varepsilon}$
Flight 3	$d_3^{1+\varepsilon}$	$d_1^{1+\varepsilon}$
Total	$2d_1^{1+\varepsilon} + d_2^{1+\varepsilon} + d_3^{1+\varepsilon}$	$2d_1^{1+\varepsilon} + d_2^{1+\varepsilon} + d_3^{1+\varepsilon}$

Table 2-1 – Total delays assigned for identical schedule

Table 2-2 compares the two allocations shown above according to their contributions to the final delay objective function. In this comparison, the two allocations are not equivalent. The value of Allocation B is less than that of Allocation A by $d_2^{1+e} - d_1^{1+\varepsilon}$. By construction, $d_2 > d_1$, and so $d_2^{1+e} > d_1^{1+\varepsilon}$. Accordingly, the objective function contribution from Allocation B is strictly less than that from Allocation A. As a result, the model will unilaterally prefer Allocation B for inclusion in the optimal slot allocation.

	Allocation A	**Allocation B**
Flight 1	$d_1^{1+\varepsilon}$	$d_3^{1+\varepsilon}$
Flight 2	$d_2^{1+\varepsilon}$	$d_1^{1+\varepsilon}$
Flight 3	$d_3^{1+\varepsilon}$	$d_1^{1+\varepsilon}$
Total	$d_1^{1+\varepsilon} + d_2^{1+\varepsilon} + d_3^{1+\varepsilon}$	$d_1^{1+\varepsilon} + d_1^{1+\varepsilon} + d_3^{1+\varepsilon}$

Table 2-2 – Final delays assigned for identical schedule

Thus, the final delay objective prefers to make any swaps that move earlier flights using fewer resources while moving later flights using more resources. Intuitively, this bias originates from the measurement of delay used – delays are counted only upon the final arrival. Because each flight's "pain" is only realized at one instance, it may be possible to swap assignments in such a manner as to achieve outcomes that, while optimal, are unsatisfactory from a variety of other viewpoints. While the mathematics of this trade are such that the values of objective function, and hence delays, are reduced, the policy implications are somewhat more difficult to discern, and will be explored further in subsequent sections.

The case shown in this section applies only when the considered flights have equal scheduled arrival times. In the following section, the more-complex cases, in which schedule order varies, are examined.

2.2.6.2 Variable schedule cases

The example shown in the previous section illustrating the bias when flights with identical schedules are considered is useful. However, it does not address the full scope of initial schedules, nor their interactions with the two feasible allocations. In this section, the remaining schedules are evaluated. It will be shown that the total delay objective function prefers to maintain schedule order, while the final delay objective function unilaterally prefers Allocation B. To this end, four variations on this initial schedule of three flights are shown in Figure 2-10.

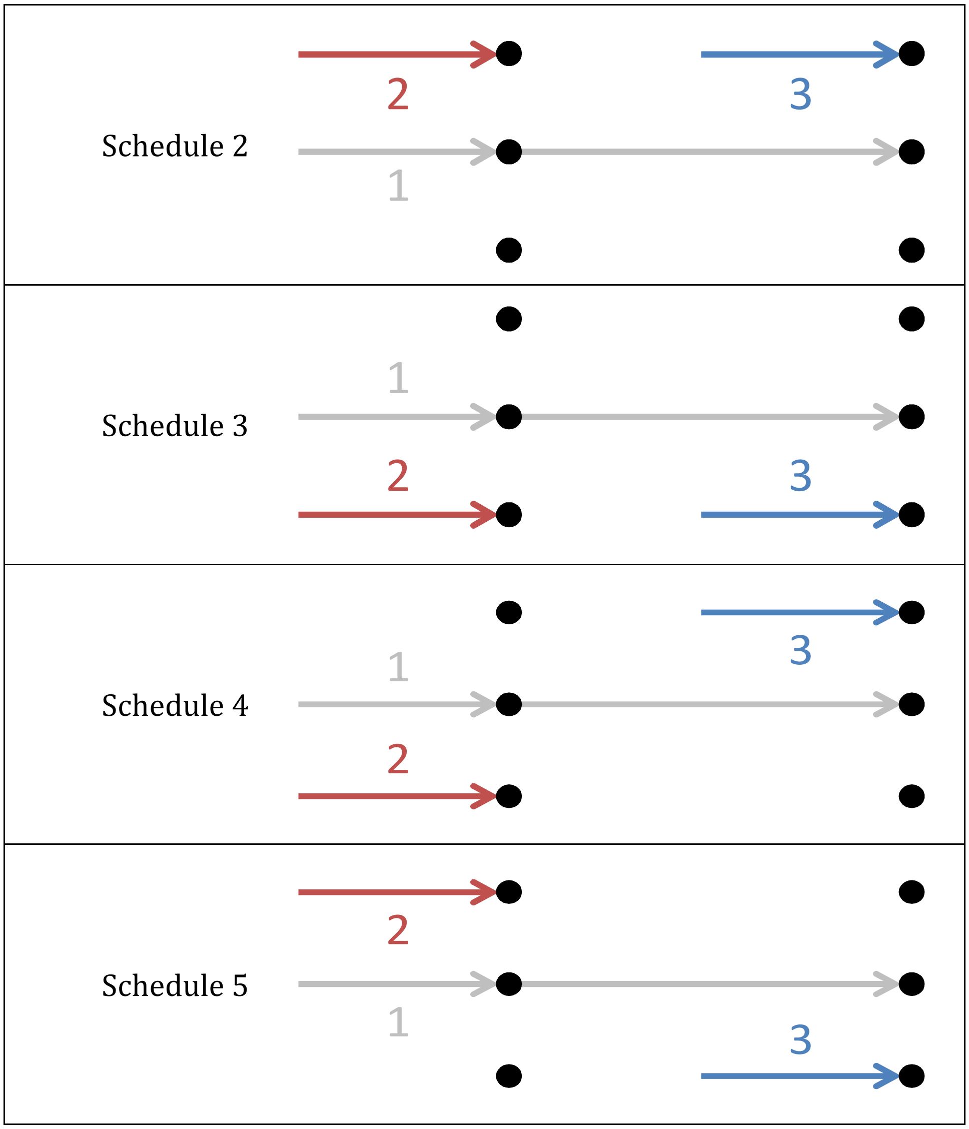


Figure 2-10 – Variable schedules

These four variations reflect all the situations in which the three flights have different scheduled arrival times. While four additional variants exist with pairs of flights sharing arrival times, they are not shown here. Those four additional cases lead to the same results, so for the purposes of brevity only these cases are shown.

Because each flight has different scheduled arrival times at each resource, the analytic exploration of changes in assigned delays becomes more complex than for the identical schedule case. To help account for this, the value q^{i} will be used to represent the delay assigned to Flight 1 at resource *i* under Allocation A. To represent the new variations in the schedule, the parameter δ^{i} will represent the scheduled headways between the two flights under consideration at resource *i*. The spacing of the slots, or the interoperation times under reduced capacities, to be allocated must also be accounted for, and will be notated as h^{i}.

The four schedules are evaluated for the total delay objective function for both allocations in Table 2-3. The relationship between the values is not as readily apparent as it was for the identical schedule case; however the trend is the same.

In general, the model's preference for one allocation or the other relies on the fact that the superlinear delay function is marginally increasing. Based on this, it is clear that the total delay objective will prefer Allocation B for Schedule 2, Allocation A for Schedule 3, and will be indifferent between these allocations for Schedules 4 and 5. Each case is consistent with the principle of maintaining schedule order.

Similar results are shown in Table 2-4 for the final objective function. In each case, the cost of Allocation B is higher than Allocation A for Flight 1. The double decrease realized by Flights 2 and 3 in going from Allocation A to Allocation B will always be greater than that increase. Therefore, Allocation B will be universally preferred, and the model will continue to favor flights using fewer resources.

Schedule 2	**Allocation A**	**Allocation B**
Flight 1	$\left(q^1\right)^{1+\varepsilon}+\left(q^2\right)^{1+\varepsilon}$	$\left(q^1+h^1\right)^{1+\varepsilon}+\left(q^2+h^2\right)^{1+\varepsilon}$
Flight 2	$\left(q^1+\delta^1+h^1\right)^{1+\varepsilon}$	$\left(q^1+\delta^1\right)^{1+\varepsilon}$
Flight 3	$\left(q^2+\delta^2+h^2\right)^{1+\varepsilon}$	$\left(q^2+\delta^2\right)^{1+\varepsilon}$
Schedule 3		
Flight 1	$\left(q^1\right)^{1+\varepsilon}+\left(q^2\right)^{1+\varepsilon}$	$\left(q^1+h^1\right)^{1+\varepsilon}+\left(q^2+h^2\right)^{1+\varepsilon}$
Flight 2	$\left(q^1-\delta^1+h^1\right)^{1+\varepsilon}$	$\left(q^1-\delta^1\right)^{1+\varepsilon}$
Flight 3	$\left(q^2-\delta^2+h^2\right)^{1+\varepsilon}$	$\left(q^2-\delta^2\right)^{1+\varepsilon}$
Schedule 4		
Flight 1	$\left(q^1\right)^{1+\varepsilon}+\left(q^2\right)^{1+\varepsilon}$	$\left(q^1+h^1\right)^{1+\varepsilon}+\left(q^2+h^2\right)^{1+\varepsilon}$
Flight 2	$\left(q^1-\delta^1+h^1\right)^{1+\varepsilon}$	$\left(q^1-\delta^1\right)^{1+\varepsilon}$
Flight 3	$\left(q^2+\delta^2+h^2\right)^{1+\varepsilon}$	$\left(q^2+\delta^2\right)^{1+\varepsilon}$
Schedule 5		
Flight 1	$\left(q^1\right)^{1+\varepsilon}+\left(q^2\right)^{1+\varepsilon}$	$\left(q^1+h^1\right)^{1+\varepsilon}+\left(q^2+h^2\right)^{1+\varepsilon}$
Flight 2	$\left(q^1+\delta^1+h^1\right)^{1+\varepsilon}$	$\left(q^1+\delta^1\right)^{1+\varepsilon}$
Flight 3	$\left(q^2-\delta^2+h^2\right)^{1+\varepsilon}$	$\left(q^2-\delta^2\right)^{1+\varepsilon}$

Table 2-3 – Total delays assigned for each variable schedule case

Schedule 2	**Allocation A**	**Allocation B**
Flight 1	$\left(q^{2}\right)^{1+\varepsilon}$	$\left(q^{2}+h^{2}\right)^{1+\varepsilon}$
Flight 2	$\left(q^{1}+\delta^{1}+h^{1}\right)^{1+\varepsilon}$	$\left(q^{1}+\delta^{1}\right)^{1+\varepsilon}$
Flight 3	$\left(q^{2}+\delta^{2}+h^{2}\right)^{1+\varepsilon}$	$\left(q^{2}+\delta^{2}\right)^{1+\varepsilon}$
Schedule 3		
Flight 1	$\left(q^{2}\right)^{1+\varepsilon}$	$\left(q^{2}+h^{2}\right)^{1+\varepsilon}$
Flight 2	$\left(q^{1}-\delta^{1}+h^{1}\right)^{1+\varepsilon}$	$\left(q^{1}-\delta^{1}\right)^{1+\varepsilon}$
Flight 3	$\left(q^{2}-\delta^{2}+h^{2}\right)^{1+\varepsilon}$	$\left(q^{2}-\delta^{2}\right)^{1+\varepsilon}$
Schedule 4		
Flight 1	$\left(q^{2}\right)^{1+\varepsilon}$	$\left(q^{2}+h^{2}\right)^{1+\varepsilon}$
Flight 2	$\left(q^{1}-\delta^{1}+h^{1}\right)^{1+\varepsilon}$	$\left(q^{1}-\delta^{1}\right)^{1+\varepsilon}$
Flight 3	$\left(q^{2}+\delta^{2}+h^{2}\right)^{1+\varepsilon}$	$\left(q^{2}+\delta^{2}\right)^{1+\varepsilon}$
Schedule 5		
Flight 1	$\left(q^{2}\right)^{1+\varepsilon}$	$\left(q^{2}+h^{2}\right)^{1+\varepsilon}$
Flight 2	$\left(q^{1}+\delta^{1}+h^{1}\right)^{1+\varepsilon}$	$\left(q^{1}+\delta^{1}\right)^{1+\varepsilon}$
Flight 3	$\left(q^{2}-\delta^{2}+h^{2}\right)^{1+\varepsilon}$	$\left(q^{2}-\delta^{2}\right)^{1+\varepsilon}$

Table 2-4 – Final delays assigned for each variable schedule case

This analysis of the properties of the two objective functions provides valuable insight into the biases that each exerts. These properties will be further explored in the case study, and will also be used to motivate the development of a rule-based solution technique in a subsequent section.

2.2.7 *Computational considerations*

In this section, several issues relating to computational implementations of this model are explored. First, the worst-case conditions for the size of the constraint matrix will be outlined. Then, two sets of valid inequalities that may improve the strength of the optimization model will be introduced. These two constraints will be introduced here and tested for efficacy in the case study at the conclusion of this chapter.

2.2.7.1 Formulation size

One measure of formulation strength and computational tractability is the size of the constraint matrix. The theoretical worst-case numbers of constraints and variables are shown in Table 2-5. These worst-case values are achieved by pathological cases using unusual combinations of slot counts and times. Several realistic numerical problem sizes are shown as well, in Table 2-6.

Entity	Worst case size of entity
Constraint set (2.4)	$\sum_{f \in F} \lvert V_f \rvert$
Constraint set (2.5)	$\sum_{i \in I} \lvert S^i \rvert$
Constraint set (2.6)	$\sum_{i \in I} \lvert S^i \rvert \sum_{f \in F} \lvert V_f \rvert$
Decision variables $\{x_{fs}^i\}$	$\sum_{f \in F} \sum_{i \in V_f} \lvert S^i \rvert$

Table 2-5 – Theoretical problem size

$\lvert F \rvert$	$\lvert I \rvert$	$\lvert S^i \rvert$	**Number of constraints**	**Number of variables**
10	2	10	440	200
100	3	150	135,750	45,000

Table 2-6 – Practical problem sizes

2.2.7.2 Backward linking constraints

The first of two valid inequalities is introduced in this section. The objective of considering this additional constraint as part of the formulation is to try to strengthen the underlying linear program. This, and the following valid inequality, will be tested computationally as to their efficacy.

This first valid inequality represents the inverse of the linking constraint presented earlier. Rather than identifying those slots to which a flight might go in the next resource, this backward linking constraint identifies those slots from which a flight must have come. This is illustrated in Figure 2-11. Exactly one of the highlighted arcs must be selected if the selected slot at resource *j* is used.

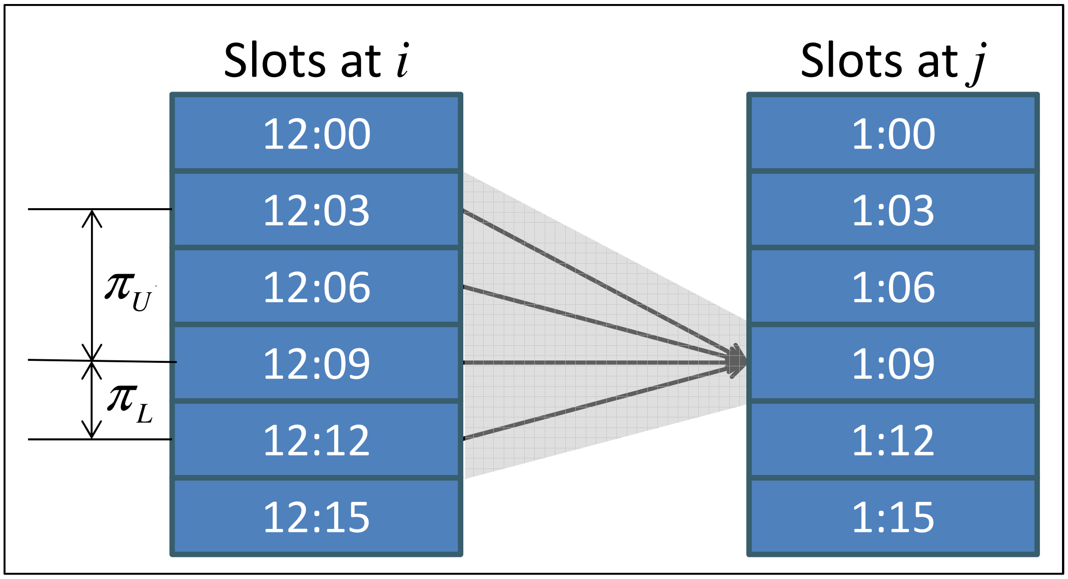


Figure 2-11 – Backward feasible range example

Similar to the construction of the forward linking constraint, the backward linking constraint is specified as shown in (2.10).

$$x_{fs}^{j} - \sum_{t\in E_{fs}^{ij}} x_{ft}^{i} \le 0 \qquad \forall f\in F, i\in V_f, j=N_f^i, s\in Q_f^i : \left|N_f^i\right| > 0 \quad (2.10)$$

The range of feasible slots E_{fs}^{ij} in the prior rationing initiative is shown in (2.11). Again, the range must account for the possibility of the slack parameter reaching past the scheduled arrival time. In this case, the lower bound of the range must be specified as the maximum of the desired range and the scheduled time.

$$E_{fs}^{ij} = \left\{k\in S^{j} : \max\left(\alpha_f^i, \tau_s^j - \alpha_f^j + \alpha_f^i - \pi_U\right) \le \tau_k^j \le \tau_s^j - \alpha_f^j + \alpha_f^i + \pi_L\right\} \qquad (2.11)$$

2.2.7.3 Summation inequality

The second valid inequality explored in this section is depicted in Figure 2-12. This constraint works by forcing equality of a sum in each pair of subsequent resources for each flight. For each feasible slot *s* for flight *f* in resource *i*, the earliest possible arrival at resource *j* can be computed. The sums from the beginning of the feasible slot range for this flight for each resource must then be equal. In the figure, this is represented by the two colored ranges.

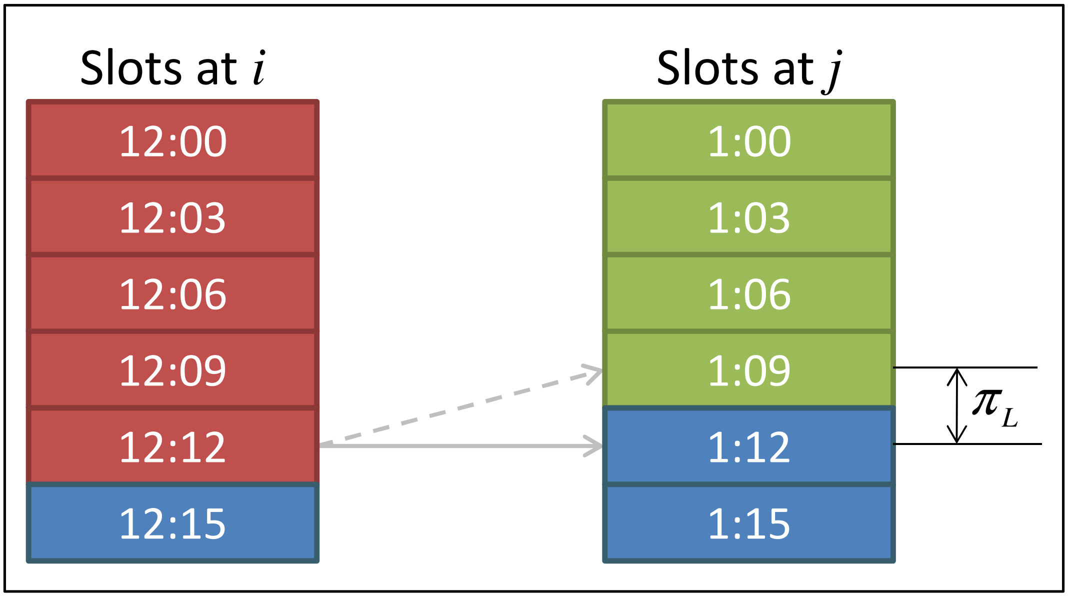


Figure 2-12 – Summation valid inequality

This constraint is enforced as shown in (2.12), indicating that the sum of decision variables representing assignments for some range in one resource must be equal to the sum of decision variables representing assignments for a related range in the next resource.

$$\sum_{r \in G_{fs}^{i}} x_{fr}^{i} \leq \sum_{t \in H_{fs}^{i}} x_{ft}^{i} \qquad \begin{array}{l} \forall f \in F, i \in V_f, j = N_f^i, s \in Q_f^i : \\ \left| N_f^i \right| > 0 \end{array} \tag{2.12}$$

The two ranges appropriate for this constraint are shown below. In (2.13), the range of slots to be summed begins at the flight's scheduled arrival time and proceeds until the slot *s* under consideration.

$$G_{fs}^{i} = \left\{ r \in S^{i} : \alpha_f^i \leq \tau_r^i \leq \tau_s^i \right\} \tag{2.13}$$

The more complex range, for the second resource, is found in (2.14). This range begins at the earliest arrival time (schedule) at resource *j*, and continues until the earliest possible arrival time, conditional on the flight having used slot *s* at the previous resource *i*.

$$H_{fs}^{j} = \left\{ r \in S^{j} : \alpha_f^j \leq \tau_r^j \leq \tau_s^i + \alpha_f^j - \alpha_f^i + \pi_L \right\} \tag{2.14}$$

2.2.8 *Practical considerations*

There are many practical considerations that should be taken into account in developing instances or in using this model in more-practical or -specific settings than those presented generally here. Each is considered briefly here in the following subsections.

2.2.8.1 Violating capacity constraints

Although this model was formulated to respect capacity constraints at each resource at being absolute, it is possible in some situations that this could or should be relaxed. In this section, modifications to the formulation are proposed to permit this by creating a second set of slots mixed within the original, each of which has a higher cost of use than the original sets S^i.

To create the framework for exceeding the nominal resource capacities, first define a new set of slots S_V^i at each resource $i \in I$. Each of these slots $s \in S_V^i$ has an associated time marker τ_s^i as before. These new slot sets and their associated times should be defined such that they overlap the previous sets, as depicted notionally in Figure 2-13.

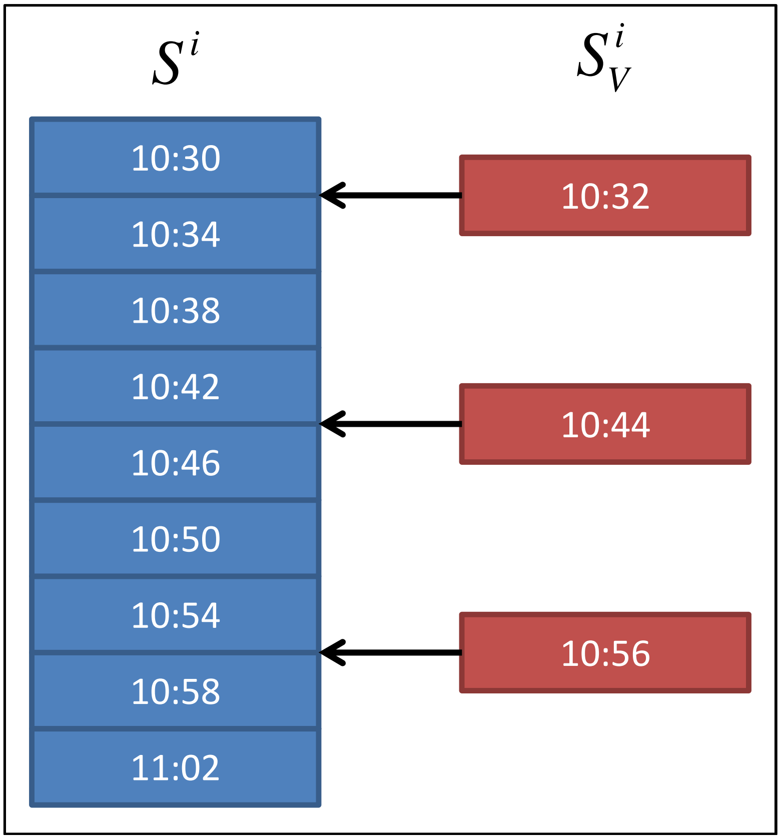


Figure 2-13 – Capacity-violating slots

To permit the assignment of flights to these new slots, the original constraints must be modified to recognize them. At each location where the set S^i is referenced, the constraint must now consider the union of the two sets $S^i \cup S_V^i$. Thus, the definition of both decision variables and constraint sets must be reworked to recognize this.

In addition, the objective function must be modified both to recognize this second shadow set of slots, and to assign higher costs thereto. The new objective proposed in (2.15) should be used. The first term represents the cost of assigning flights to those slots in the original standard set. The second term considers only the cost of assigning flights to slots in this new set, and assigns a cost differential χ to account for the capacity violation.

$$\min z = \sum_{f \in F} \sum_{i \in V_f :} \sum_{s \in Q_f^i \cap S^i} \left(\tau_s^i - \alpha_f^i \right)^{1+\varepsilon} x_{fs}^i + \sum_{f \in F} \sum_{i \in V_f} \sum_{s \in Q_f^i \cap S_V^i} \chi \left(\tau_s^i - \alpha_f^i \right)^{1+\varepsilon} x_{fs}^i \qquad (2.15)$$

2.2.8.2 En route flights

The first boundary condition addresses those flights already airborne when this optimization model is run. Because they are already airborne and have planned for a certain route and arrival time, it would be inappropriate, except in very serious conditions, to assign considerable delays or deviations from their plan. Accordingly, and consistent with operational practice, they are exempted from the controls exerted by such capacity rationing programs.

In principle, these flights could be included in the instance, and their appropriate decision variables fixed to a value of one for the slots corresponding to their current plan. However, this increases the size of the formulation while adding little value. Instead, the slots at their respective arrival times will simply be removed from consideration by the model.

Figure 2-14 depicts an example case of exempted en route flights. Assume that the slot list shown corresponds to some resource, either an airport or an airspace region, and that the flights shown are already en route when planning begins for the coordinated capacity rationing. The arrows indicate the planned arrival times to this resource for each of these flights. Because of the limited flexibility these flights have, the time slots corresponding to their planned arrival times are marked and simply removed from consideration by the model as part of the preprocessing routines.

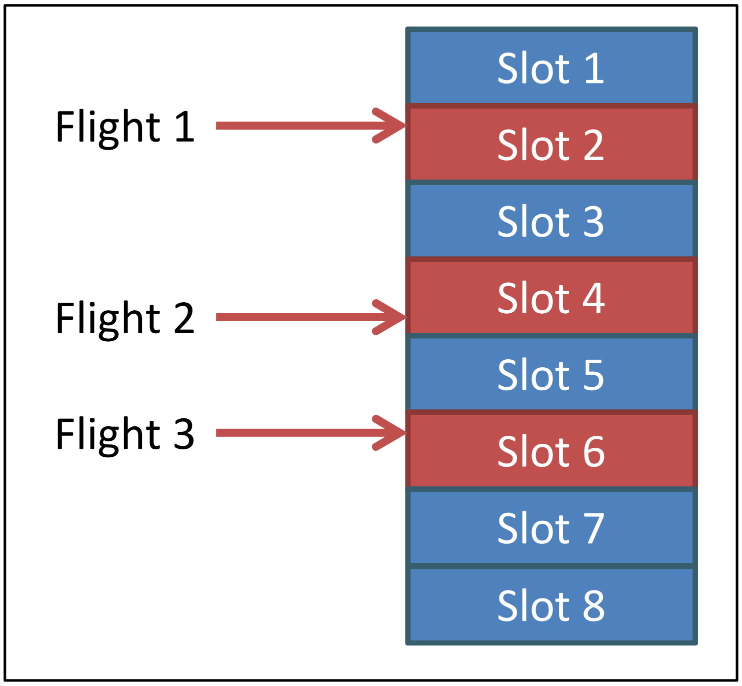


Figure 2-14 – Sample flight exemptions

It is possible that this approach may lead to providing sufficient capacity for the number of exemptions required, if both the capacity drop is sufficiently large and enough flights are already en route. If, rather than removing the slots, the flights are fixed using decision variables, then the optimization algorithm itself will quickly detect this condition as an infeasibility. If the approach of removing slots is used as is suggested here, then the modeling process should include a preprocessing step that checks whether sufficient capacity exists for the exempted flights. If infeasible conditions are detected, then the model should recommend serious remedies, including assigning airborne delays and flight diversions.

This confluence is unlikely, however, because of the lead times typically provided by forecasts of future capacity availability. The number of flights already en route, and hence unable to revise significantly their planned arrival times, varies inversely with the lead time used in planning the capacity rationing. For example, if the planning process begins two hours before the capacity drop is expected to occur, then the number of flights already en route will be small indeed, and only in the most extreme of circumstances will this violate the new bound. Because such events should be extremely rare, they will in general be ignored in modeling this problem. However, for those few flights that are already en route, the mechanism presented in this section may be employed.

2.2.8.3 Flights scheduled immediately before rationing

Flights scheduled to arrive immediately before a capacity rationing program is set to begin represent a challenge in this model if they are delayed beyond their planned arrival time sufficiently so that their new arrival time falls within the time bounds of the program. In this case, their delayed arrival implicitly includes them in the rationing program. This may force flights explicitly included to experience airborne delays to comply with flight spacing requirements. In sufficient numbers, flights spilled over into a program such as this may force a revision to the initial plan, which of course has greater impacts when considering connected resources. However, this would be a fairly rare phenomenon. Further, sufficient slack is typically included in planning these capacity rationing programs to allow minimal spill from earlier time periods. As a result, this will be ignored.

2.2.8.4 Planning lead time

An important issue in developing instances and in using the results of this optimization model is the lead time provided for decision making. This problem is not unique to the coordinated scenario addressed here, but impacts single resource problems as well.

In general, plans of this nature could be made at any time and implemented almost immediately. However, the quality of the weather forecasts used to generate resource capacities varies inversely with the lead time used in making the forecast. Conversely, implementing such decisions as early as possible provides the greatest amount of freedom, as more flights remain on the ground as the lead time increases.

Thus, it is prudent to wait until good enough information is available, but while sufficient freedom exists to implement a useful capacity rationing plan.

2.2.8.5 Setting program length

Another important practical issue is setting the length of time during which capacity rationing is taking place at each resource. Once again, this is not a problem unique to the coordinated case, but also impacts single resource models as well.

Assuming a simple model of capacity, as depicted notionally in Figure 2-15, there are several candidates for the end time of capacity rationing. The first of these would correspond to some time before t_1, or before the end of the capacity disruption. This is clearly not sensible and is contrary to the objectives of this work, and so is not considered here.

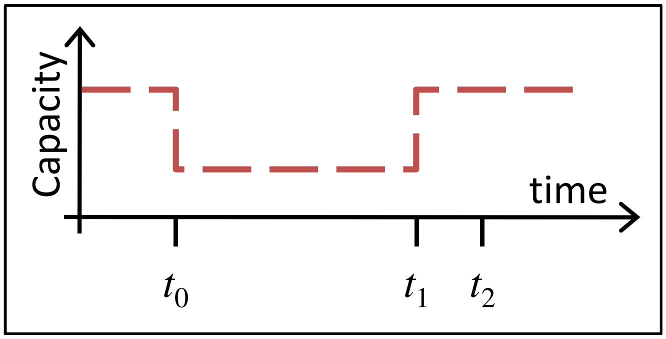


Figure 2-15 – Simple model of capacity

Two other cases are examined here: ending the capacity rationing program at the expected end time of the disruption (t_1) or at some time after that (t_2). If the schedule during the disruption is sufficiently sparse, then ending at t_1 may be sensible. However, this is unlikely. As a result, one must determine the precise value of t_2. This is a difficult problem, and it must incorporate both the schedule density as well as the uncertainty associated with the capacity forecast. The uncertainty is truly the key issue in this case, as the end time is potentially a long time from the time at which the capacity rationing procedure is being undertaken.

2.2.8.6 Simultaneous planning and dynamic execution

An important assumption taken, mentioned earlier, in modeling this problem is that an integrated plan is developed simultaneously for all resources under consideration. For coordination between resources to have some meaningful impact, this strong assumption is important. Of course, in practice, simultaneously and confidently identifying several resources expected to experience congestion is difficult. If one resource is identified as confidently expected to experience congestion and another less so, one may try to wait out the second resource to determine if intervention will be needed. The difficulty in this approach is that the freedoms available for rationing at the first resource dwindle as time advances. Thus, it is clear that some system must be available for initial planning with the possibility of revisions.

The simplest method to conduct these revisions is by incorporating the en route flight paradigm described in the first of these subsections. Using that, flights that have already departed under the previous capacity rationing plan will be exempted from revisions. Then, the new resources and flights can be added to the instance, and the entire problem can be solved again. In principle, the solution should be simpler to find because potentially many of the variables will be fixed by the en route conditions.

Revisions may also be necessary if a more complex model of capacity evolution is adopted under which conditions may deteriorate. In this situation, this model can continue to be employed, but the slack parameters used in making the linked assignments may need to increase to allow for the assignment of airborne delays. In this situation, it may also be useful to employ some heuristic method for assigning revised delays. In any case, if there is a reasonable suspicion of needing a revision, then explicitly including this information in the model may be productive. A stochastic formulation to address this is described in §4.

2.2.8.7 The ghost in the machine

The final practical concern described in this section describes the challenges of employing optimization on practical problems. Optimization models find solutions based only on the feasible region and the costs as they have been defined. They do not "know" whether a variable represents assigning a flight to a time slot, ordering 1,000 more widgets, selecting a new location for a construction project, or any other decision. The constraints and costs are simply mathematical expressions, and could represent any number of systems.

One of the primary symptoms of this blindness that optimization models experience is their lack of transparency. In one sense, a set of constraints and an objective will yield one of a set of optimal solutions, if such exists. However, the means by which this solution is derived are somewhat opaque to users. This may introduce a multitude of unintended consequences. An excellent example of these unintended consequences was demonstrated previously, with the discussion of the biases introduced by each objective function. While a user may expect that each of these objectives will yield different solutions, the means by which the optimization model arrives at them is rarely immediately apparent.

Further, in a practical and distributed setting such as air traffic management, it is inconceivable that every user, including airlines and private aviation, should maintain a separate installation of complex and expensive optimization software, along with the personnel to use it. Further, given that the input data must be processed such that they are compatible with the optimization system, additional layers of software beyond the solver libraries themselves are needed.

Finally, even if all users have agreed to use an optimization methodology for managing some system, there are no, or limited, guarantees that the models will yield optimal, or even feasible, solutions quickly. Experimental evidence will show that it is typical to achieve good solutions quickly; however, this is not as a rule true.

Each of these factors has limited the application of optimization models in operational air traffic systems. While an academic setting affords the possibility to carefully analyze the unexpected consequences of an optimization model, in a distributed operational setting, it is preferable that processes be transparent to and easily replicable by all participants.

2.2.9 *Formulation summary*

In this subsection, the formulation is repeated for clarity.

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| = 0}} \sum_{s \in Q_f^i} \left(\tau_s^i - \alpha_f^i\right)^{1+\varepsilon} x_{fs}^i \tag{2.9}$$

$$\sum_{s \in Q_f^i} x_{fs}^i = 1 \qquad \forall f \in F, i \in V_f \tag{2.4}$$

$$\sum_{\substack{f \in F : \\ i \in V_f}} x_{fs}^i \leq 1 \qquad \forall i \in I, s \in S^i \tag{2.5}$$

$$x_{fs}^i - \sum_{k \in R_{fs}^{ij}} x_{fk}^j \leq 0 \qquad \forall f \in F, i \in V_f, j = N_f^i, s \in Q_f^i : \left|N_f^i\right| > 0 \tag{2.6}$$

$$x_{fs}^i \;\; binary \qquad \forall f \in F, i \in V_f, s \in Q_f^i \tag{2.2}$$

The above equations represent only one version of this formulation. Alternatively, the final delay objective function (2.9) may be replaced with the total delay objective function shown in (2.8).

$$\min z = \sum_{f \in F} \sum_{i \in V_f} \sum_{s \in Q_f^i} \left(\tau_s^i - \alpha_f^i\right)^{1+\varepsilon} x_{fs}^i \tag{2.8}$$

2.3 Rule-based capacity rationing

In an effort to obviate the concerns expressed in the previous section about using optimization for a practical problem in a distributed system, this section introduces a rule-based procedure to address the same problem. Many of the above practical considerations represent fixes for the unintended consequences of using an optimization model. Rather than build a model within a flexible environment, in this section, a solution methodology described is designed specifically for the problem at hand. A rule-based procedure is described here to help allay concerns about transparency and complexity in employing optimization models for practical situations.

The primary advantage of this rule-based procedure is that it explicitly provides transparency. The rules by which each decision and allocation occur are specified and are easily comprehended. Additionally, the rules and data structures needed for these procedures can be coded in a variety of computational environments, including spreadsheets. This provides a high degree of availability to many user groups, and greatly lowers the entry barriers to active participation in the process. Also, execution times for these procedures will be very short. Finally, one of the procedures proposed in this section is designed explicitly to mimic the results of the final delay optimization model. Developing an algorithmic approach that can achieve results comparable to those from an optimization model is a valuable achievement in itself.

The next portion of this section discusses the heuristic itself and defines a procedure that it calls for flights using multiple resources. These are introduced generically, so the section concludes with two different approaches to defining the parameters of the heuristic.

2.3.1 *Outline of procedures*

In this section, two heuristics are formally presented. The first, the rule-based capacity rationing (RCR) procedure, is the critically important portion of this section. It defines the procedure by which flights are prioritized and allocated to slots. If only flights using a single resource are considered, then RCR represents simply a generalization of the widely employed Ration by Schedule (RBS) procedures. However, it is the second procedure, called by the first, which brings the coordination problem into this section. The multi-resource feasible slot identification (MFSI) procedure is used to identify feasible slot combinations for flights using multiple resources.

2.3.1.1 Rule-based capacity rationing procedure

At a high level, this procedure is a greedy heuristic for feasibly allocating flights to slots. Flights are sorted according to several criteria. They are then assigned iteratively to the best available slots that are feasible for their operations. Conditions for feasibility include slot times later than planned/scheduled arrival times at each resource and compatibility with slots at adjacent resources. The procedure is parameterized to allow different priority rules to be employed. It is presented for the case in which the maximum number of resources visited, $\left|V_f\right|$ is two or less. The generalization is straightforward, but is not presented for simplicity. To track available slots, define an indicator variable for each slot *t* at each resource *j* I_t^j that takes on a value of one when a flight has been assigned to it and zero otherwise.

Procedure RCR

1. Sort flights using several keys to create an ordered list Φ of flights. Sufficient keys must be used to ensure that there are no ties.
2. Remove the first flight from the list Φ and call if *f*.
3. Identify the first resource *i* visited by flight *f* according to $i=\left\{i\in V_f:\alpha_f^i=\min\limits_{l\in V_f}\left(\alpha_f^l\right)\right\}$.
4. Find the set of available and feasible slots Σ*^i^* at resource *i*, according to $\Sigma^i=\left\{s\in S^i:I_s^i=0,\tau_s^i\ge\alpha_f^i\right\}$.
5. Identify and remove the earliest slot *s* from Σ*^i^*, such that $s=\left\{s\in S^i:\tau_s^i=\min\limits_{t\in\Sigma}\tau_t^i\right\}$.
6. If $\left|V_f\right|=1$, then assign flight *f* to slot *s* at resource *i*, set $I_s^i=1$.

7. If $\left|V_f\right| > 1$, then execute procedure MFSI. If it identifies a feasible slot *t* at the next resource $j = N_f^i$, then assign flight *f* to each of these slots, and set $I_s^i = 1$ and $I_t^j = 1$. Otherwise, if MFSI cannot identify a feasible slot at resource *j*, given that flight *f* uses slot *s* at resource *i*, then another candidate slot must be tested, so go to step 5.
8. If $\Phi = \varnothing$ is empty, then end. Otherwise, go to step 3.

It is possible that this allocation procedure could yield slots to which no flight is assigned. It is likely, depending of course on the instance under consideration, that there will be unassigned slots at the end of the sequence. This is of little concern, as they present no loss to efficiency. However, empty slots in the midst of others that are assigned present the possibility that the heuristic has found a bad allocation. These gaps will only occur because no feasible slot combinations for flights using multiple resources could make use of that orphaned slot. Otherwise, all flights are assigned the earliest possible available and feasible slot. As a result, these empty slots should be quite rare, and will not be addressed through a separate swapping procedure.

2.3.1.2 Multi-resource feasible slot identification

The most complex portion of the above procedure is identifying compatible slot pairs for flights using multiple resources. To simplify the exposition of the generic procedure, this is presented separately here. The procedure takes as input some candidate slot *s* has been identified at some resource *i* for flight *f*. For simplicity, assume that *i* is the first of two resources visited by flight *f*. The procedure is applicable to longer sequences, or to resources on the interior, or at the end of the sequence, but these cases are not presented. Denote the next resource in the sequence after *i*, N_f^i, as *j*.

As an example to introduce the procedure, examine Figure 2-16. The procedure works for scanning from available slots at resource *i* for feasible and available slots at resource *j*.

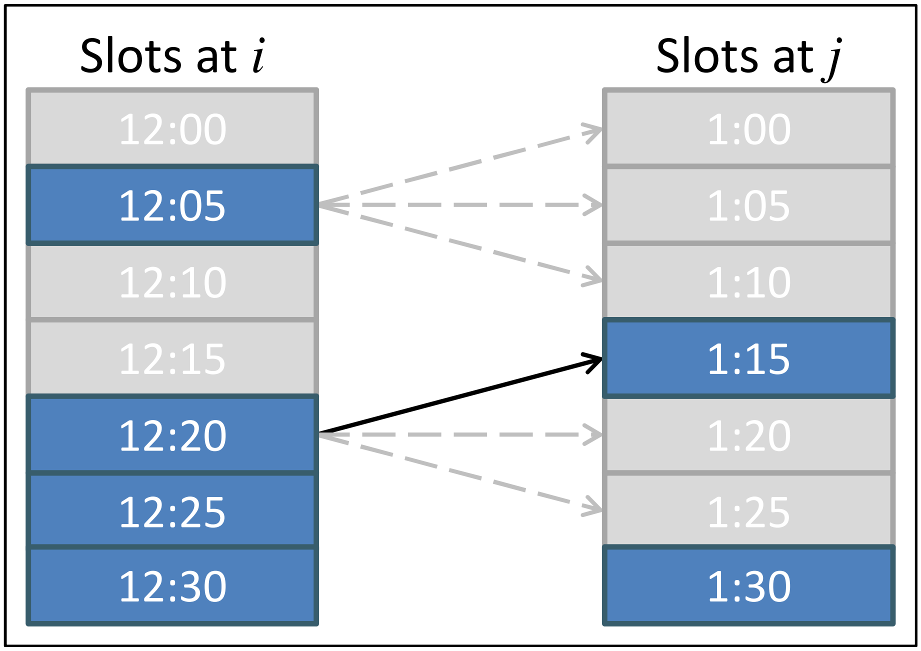


Figure 2-16 – MFSI example

In this example, assume that the travel time between the two resources is 1 hour and that the maximum deviation allowed in linked assignments is 5 minutes in each direction. It would begin with the 12:05 slot at resource *i*, but upon scanning resource *j*, find no compatible slots. Accordingly, it would next examine the 12:20 slot. In that case, the 1:15 slot at j is available and within the appropriate time range, so these two slots will be returned as a compatible pair. The other compatible pairs that it would identify through further scanning are (12:25, 1:30) and (12:30, 1:30).

Procedure MFSI

1. Identify the set of feasible and available slots Z^j at resource *j*, according to $Z^j = \left\{ t \in S^j : I_t^j = 0, \tau_t^j \geq \alpha_f^j \right\}$

2. Calculate the time difference between each of these slots in $t \in \mathrm{Z}^j$ and the preferred arrival time at resource *j*, $\delta_{st}^{ij} = \tau_t^j - \left(\tau_s^i + \alpha_f^j - \alpha_f^i\right)$
3. Find the minimum absolute value of this array according to $\delta^* = \min\limits_t \left|\delta_{st}^{ij}\right|$, and the slot index t^* associated with this minimum deviation, as in $t^* = \left\{t \in S^j : \left|\delta_{st}^{ij}\right| = \delta^*\right\}$.
4. Compare the deviation $\delta_{st^*}^{ij}$ associated with this slot t^*, to the range established by the maximum deviation parameters π_L and π_U, according to $\delta_{st}^{ij} \underset{?}{\in} \left[-\pi_U, \pi_L\right]$. If this condition shown is true, then the slot t^* is feasible. Otherwise, no feasible slot at resource *j* exists for flight *f*, given that it uses slot *s* at resource *i*.

This heuristic procedure for identifying feasible slot pairs at subsequent resources will find the earliest combination that exists at both resources. In the next two sections, specific parameters for prioritizing flights as input to the RCR procedure are defined.

2.3.2 *Final delay procedure outline*

The first rule-based approach to coordinated capacity allocation builds on the analytic discussion of the final delay objective shown previously. It was demonstrated that this objective will move as late as possible flights using multiple resources, in an effort to move earlier in time flights using fewer resources. This principle is used here to derive a rule-based approach for developing coordinated capacity allocations. This procedure is named Final Delay Priority (FDP).

The procedure RCR from above is used with the following sort keys. The first two should break most ties, and the third is guaranteed to break any that remain.

- Key 1: Number of resources visited by flight f, $\left|V_f\right|$ (increasing)
- Key 2: Scheduled arrival time of flight *f* to its destination, $\alpha_f^i = \max\limits_{k \in V_f} \alpha_f^k$ (increasing)

- Key 3: Tail registration number (N number) of the aircraft used to operate flight *f* (increasing)

2.3.3 *Resource priority procedure outline*

An alternative rule-based procedure for creating coordinated slot allocations is to allocate capacity at specific resources first, rather than focusing on flight characteristics. This policy replicates the procedure used in practice, wherein allocations at airports take precedence over those at airspace resources. This procedure is named Resource Allocation Priority (RAP).

For this case, some preferred ordering of the resources themselves must be provided, called Γ. If airports are to be prioritized, then all airports will precede airspace resources in this ordering. Global sort keys are used per se – the ordering of the flights must be developed iteratively, as described below. Thus, because of the alternative sort procedure, when implementing RAP, these steps replace step 1 in the RCR procedure.

1. Remove the first resource from the list Γ and call if *i*.
2. Identify the subset of flights that visit *i* at some point during their route, according to $\Phi^i = \left\{ f \in F : i \in V_f \right\}$.
3. Sort the list Φ^i according to the following sort keys:
    - Key 1: Scheduled arrival time of flight *f* to resource *I*, α_f^i (increasing)
    - Key 2: Tail registration number (N number) of the aircraft used to operate flight *f* (increasing)
4. Append the sorted list Φ^i to the end of the master list of flights Φ
5. If $\Gamma = \emptyset$ is empty, then end. Otherwise, go to step 1.

2.4 Case study

The efficacy of the both the optimization and rule-based approaches proposed in this chapter are explored here through a case study. Schedule data are randomly generated, but represent a realistic situation such as is encountered during summer convective weather over the northeastern United States. First, the physical and temporal characteristics of the case study will be described, then several categories of results concerning overall model performance, equity, and computational issues will be evaluated.

2.4.1 *Input data*

Essential to the results of this case study are the input data used to drive the models. This section describes the physical and schedule-related data used.

It is important to note that artificial, but realistic, data are used. This is primarily motivated by the myriad challenges present in acquiring, cleaning, and processing historical records. Because much of the required data are not generally publicly available, simply identifying a day as a case study is very challenging. Even if a suitable time period is identified, flight records are often incomplete or corrupted. Capacity data, as they are envisioned by these models, are particularly difficult to obtain as well. For these reasons, randomized procedures are used to generate flight routes and schedules in this work. They will be described where applicable.

2.4.1.1 Physical configuration

A simple physical configuration was chosen for this case study. There are two airports (B, C) for which capacity is being rationed, and one disruption (A) in the en route airspace for which rationing must take place, as depicted notionally in Figure 2-17. This notional layout is comparable to the real case shown in Figure

1-10 for the northeastern portion of the United States. Travel times between resources are taken as a constant 60 minutes for each flight.

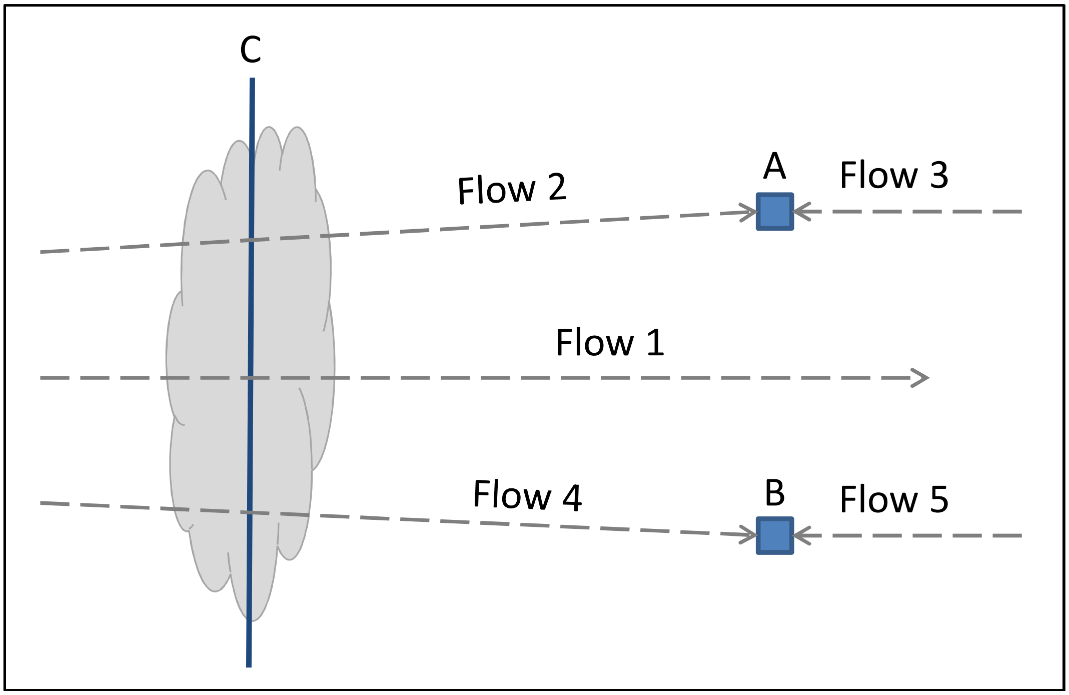


Figure 2-17 – Case study layout

To better visualize the problem setup, some results will be discussed in terms of flows, as labeled in this figure. Flow 1 comprises flights crossing the en route disruption, but not traveling to either of the two disrupted airports. Flows 3 and 5 travel to Airports B and C, respectively, but do not cross the en route disruption. The most interesting flows, 2 and 4, cross the disrupted airspace before arriving at the disrupted airports (B and C, respectively). It is these two groups of flights that confound the traditional single-resource rationing methods.

2.4.1.2 Flight schedules

Flight schedules are the primary driver of complexity in these problems. For this case study, these are generated randomly using the procedure outlined below.

1. Generate uniform schedule using full capacity for each resource: For this case study, the airspace resource has a capacity of C_A flights per

hour, while each airport may accept at most C_B and C_C flights, respectively.

2. Randomly remove a fraction of the uniform schedule: These airports are assumed to be fairly busy, so each flight is removed from the uniform lattice with probabilities p_B and p_C. Flights are more likely to be removed from the airspace resource, at probability p_A.
3. Choose fraction of airport flights to use multiple resources: Of the remaining flights at each airport, each is given a f^M probability of also using the airspace resource. Those chosen by this method are assigned an arrival time to the airspace resource corresponding to the correct inter-resource travel time t_{ij}. Adding them to the airspace resource schedule in this fashion requires the removal of a large fraction of flights from the uniform schedule, as specified in the previous step, to avoid an overly congested airspace schedule.
4. Randomly generate flight length according to some probability density function: Although the data are not used explicitly in this case study, for completeness, flight departure times are also calculated.

For the procedure described above, the parameter values shown in Table 2-7 were used. A discrete distribution shown in Figure 2-18 is used to generate flight lengths. Each flight samples from this distribution to choose a length range. Within the bounds of that range, a precise flight length is generated randomly by sampling from a uniform distribution.

Name	Description	Value
C_A	Nominal capacity of Resource A	60
C_B	Nominal capacity of Resource B	50
C_C	Nominal capacity of Resource C	50
p_A	Probability for removal at Resource A	0.5
p_B	Probability for removal at Resource B	0.2
p_C	Probability for removal at Resource C	0.2
f^M	Fraction of multi-resource flights	0.4
t_{AB}	Travel time from A to B (minutes)	60
t_{AC}	Travel time from A to C (minutes)	60

Table 2-7 – Scheduled generation parameter values

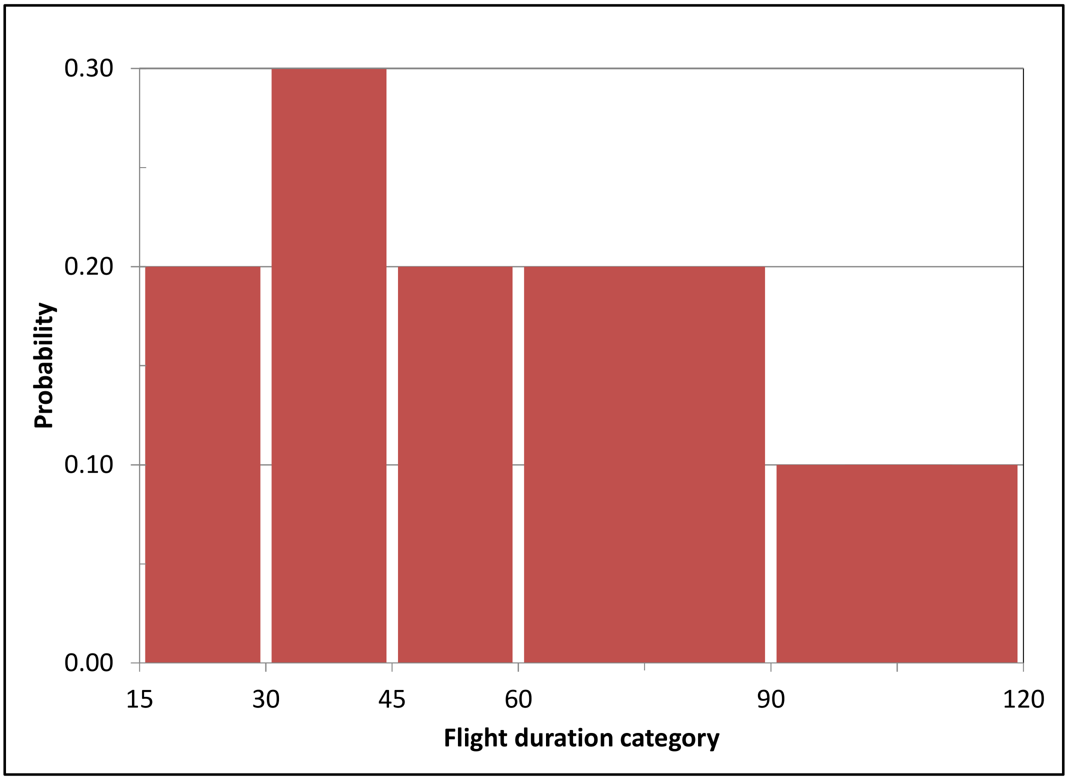


Figure 2-18 – Distribution of flight length ranges

After the above procedure was completed, a complete schedule for all flights considered was generated. The number of scheduled arrivals to each resource over the study period is shown in Figure 2-19. Because the travel time between the storm and each airport is 1 hour, the bars representing the schedule for Flows 2 and

4 are simply shifted by 1 hour from their appearance in the Resource A schedule to their appearances in the Resource B and C schedules, respectively.

The schedule is assumed to terminate after the flights shown in Figure 2-19. While potentially unrealistic, this simplifies considerably the conditions surrounding the end of the program because the flights expected to arrive after the end of the program are not subject to rationing.

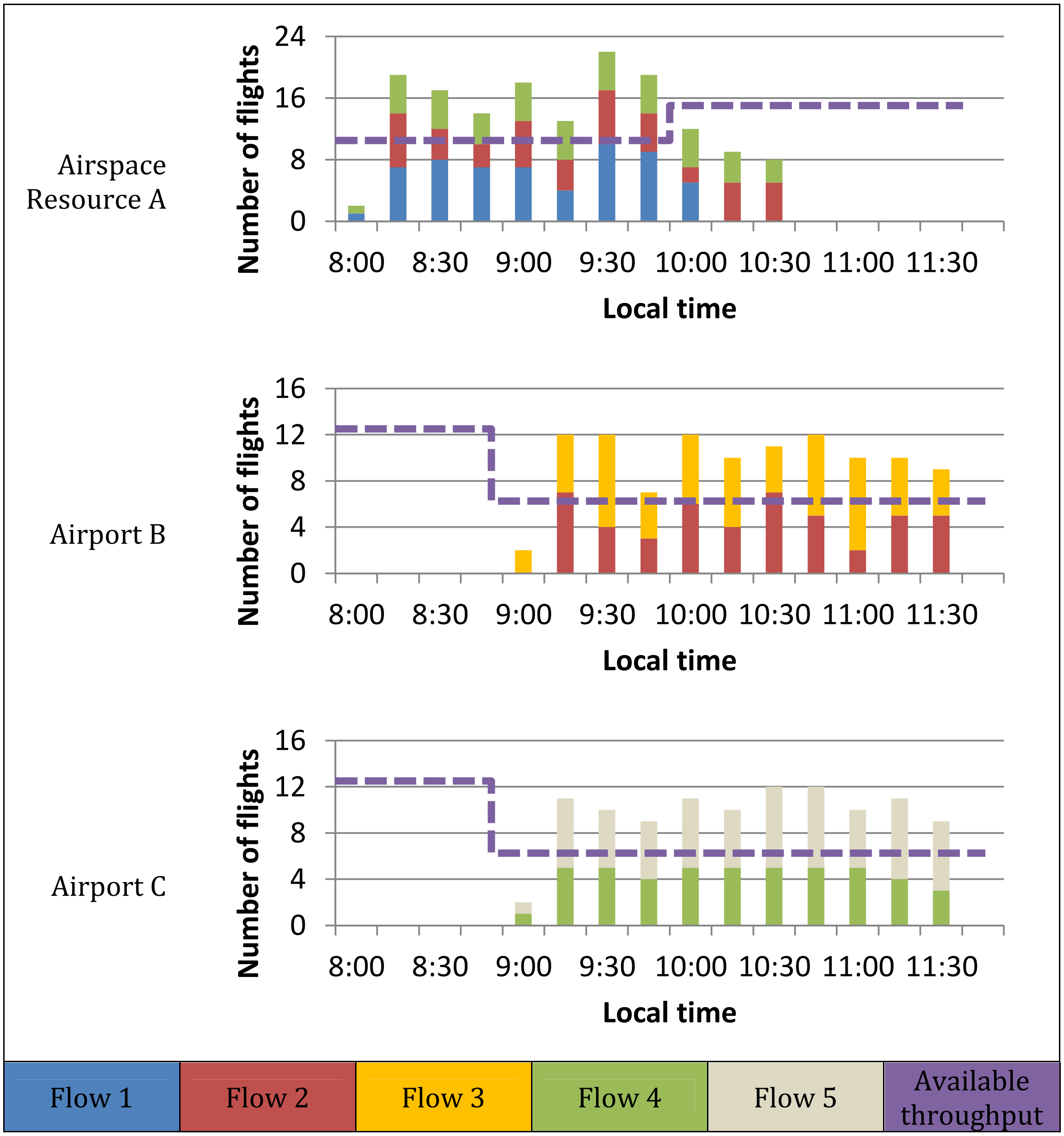


Figure 2-19 – Nominal resource schedules

Because the capacity reduction is sufficiently extreme relative to the scheduled number of aircraft, the optimization model will assign flights to nearly every slot. Thus, the time-varying profile of flights after the model has run will match precisely with the reduced capacity line until the entirety of the set of flights has been assigned.

2.4.2 *Computational testbed*

The computational experiments in this case study were performed using powerful computer hardware and software. The system used has four dual-core Intel Xeon X5355 processors and 12GB of memory. It runs software in a 64-bit environment under Windows Server 2003 Enterprise edition.

The optimization tests were conducted using Fair Isaac's Xpress 2008b 64-bit software. Models were coded using the Mosel language and executed through Xpress' graphical interface, Xpress-IVE. The rule-based approaches were coded in MATLAB R2008a running on the same hardware.

2.4.3 *Justification of approach*

Before going into details about the allocations generated by the various solution methodologies proposed in this chapter, it may be useful to demonstrate empirically that the need for coordination exists in realistic case studies.

The most basic method by which the need for the method proposed here may be evaluated is to determine the number and severity of infeasibilities induced by the independent allocation process. To this end, the independent Ration By Schedule allocation for each resource was determined. For those flights using two resources, the travel time required by these independent allocations was compared to then nominal 60 minute travel time. A histogram of these deviations from nominal is shown in Figure 2-20. The bars corresponding to those flights that would have been feasible under the speed-up/slow-down assumptions of the model are depicted in red. Of the flights that used multiple resources, only 40% would have received a feasible allocation from the independent process, while 60% would not. Deviations of up to -17 minutes were indicated.

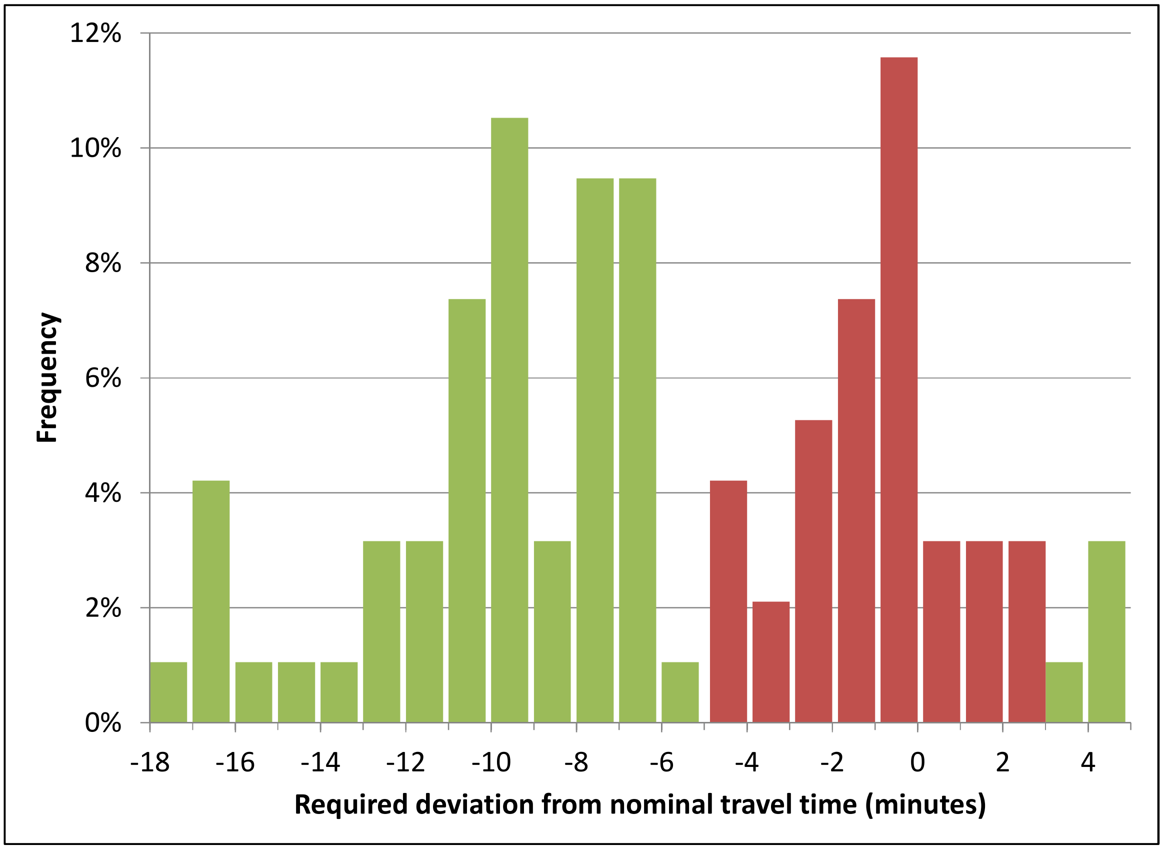


Figure 2-20 – Feasibility analysis for case study

The results of the above analysis indicate clearly that considering resources independently cannot generate feasible capacity allocations for the majority of the included flights. To that end, the next section compares the approach of simply prioritizing resources against the optimization models.

2.4.4 *Aggregate comparison of assigned delays*

The first issue evaluated in this case study is the relative advantage in allocating delays provided by the various approaches proposed here over the independent RBS process. This analysis ignores the issue demonstrated above that the independent RBS allocations are not feasible for a significant proportion of flights. In Table 2-8, the total amount of delays associated with every resource, as well as the arrival delays associated with every flight are shown for several allocation methods.

The base formulation with each of the two objective functions is evaluated. The rule-based method that prioritizes flights based on the number of resources they visit (FDP), and the airport priority method (RAP) are also considered. Objective function values are not compared. The two optimization models sum different terms, and there is no reason to believe that their relative magnitudes should be related. In addition, the rule-based methods do not yield objective function values per se.

Solution methodology	**Total delay assigned (minutes)**	**Arrival delay assigned (minutes)**
Independent RBS	**14169**	10798
Total objective	14169	10723
Final objective	14394	8927
FDP	19947	10215
RAP	14305	10585

Table 2-8 – Aggregate comparison of allocation

The first line in this table indicates the total amount of "delay" assigned at each resource by the independent RBS process. By construction, this quantity represents the minimum value of total delay that can be allocated. However, because only some of these "delays" are realized, this value is strictly larger than the amount of arrival delay realized by flights. As a result, there is no reason to suspect that this allocation should minimize arrival delays. In fact, it does not. Several other allocations, most notably the optimization model with the final delay objective, have lesser amounts of arrival delay. These results indicate that it is possible to derive allocations that minimize total assigned delay, as with the independent RBS process.

However, it is also possible to allocate lesser arrival delays through other means. These other mechanisms have the potential to improve system performance and will be examined now in greater detail.

The final row, RAP, is an approximation of the approach employed in practice of prioritizing the airport allocation. According to these metrics, this allocation does improve upon the independent process, as well as the total delay optimization model. However, it assigns considerably greater total arrival delays than does the final delay optimization model and its proxy, the FDP.

An additional aggregate comparison of results is shown in Table 2-9. In this case, the optimization results are compared to the heuristic results at several time milestones. This provides a snapshot of the computational performance of these models.

Metric	**Solution time**	**Solution methodology**			
		Total	*Final*	*FDP*	*RAP*
Total delay (minutes)	2 minutes	14169	14169	19947	14305
	30 minutes	14169	14554		
	Optimal	**14169**	14394		
Final delay (minutes)	2 minutes	10703	10723	10215	10585
	30 minutes	10718	10333		
	optimal	10723	**8926**		

Table 2-9 – Comparison of assigned delays

One important caveat in evaluating these aggregate results is that those for the final delay objective function model are not optimal. The solution algorithm was terminated after 36 hours with a gap from the best available bound of 0.44%.

The most apparent trend is that each of the optimization methods, total and final delay objectives, minimizes their respective associated metrics at optimality, as shown by the bolded cells in the table. This result is expected, but confirms the validity of each. Neither of the rule-based methods achieves an allocation of equal quality.

Another important property displayed in the table above is the evolution of the solutions from the optimization models. The total delay model seems to yield an optimal, or nearly optimal, solution very quickly. On the other hand, the final delay model shows a great difference between the initial values of delay assigned versus the optimal amount. This evolution will be explored in greater detail in §2.4.6.

2.4.5 *Comparison of treatment of various flows*

The next analysis of these allocations considers a different metric of the distribution of flight delays. The first aggregation, in terms of flight destination, is shown in Table 2-10. These results are interesting because they demonstrate that three of the four solution methods find allocations of equal efficiency for the airport resources, B and C. However, the distribution of delays at these resources clearly differs, as indicated by the various standard deviation values shown. The FDP method induces some inefficiency at the airports, as indicated by the increased amount of delays assigned there. However, the intent of this table is to indicate again that the final delay optimization model minimizes delays by reducing them for those flights using a single resource, in this case, airspace resource A.

Dest.	**Number of flights**	**Mean arrival delay (standard deviation)**			
		Total	*Final*	*FDP*	*RAP*
A	58	32.0 (14.6)	1.1 (0.7)	1.1 (0.7)	29.7 (11.8)
B	107	42.4 (19.1)	42.4 (25.1)	49.0 (54.2)	42.4 (18.8)
C	107	40.5 (19.1)	40.5 (25.4)	45.9 (52.1)	40.5 (19.0)

Table 2-10 – Comparison of average delays by destination

Of course, the analysis of the optimization objectives shown previously indicated that, for a single destination, the distribution of flights may vary significantly. Using the flows defined in Figure 2-17, aggregated delays are shown in Table 2-11. Although the flights within each flow are distributed over time, the fact that they use the same sequence of resources suggests that an equitable model would treat each of these flights within the flow equally.

Flow	Number of flights	Mean arrival delay (standard deviation)			
		Total	*Final*	*FDP*	*RAP*
1	58	32.0 (14.6)	1.1 (0.7)	1.1 (0.7)	29.7 (11.8)
2	48	40.1 (17.4)	58.3 (16.5)	105.1 (28.0)	42.5 (19.1)
3	59	44.2 (20.4)	29.4 (23.3)	3.4 (2.1)	42.3 (18.8)
4	47	38.1 (17.8)	55.3 (20.2)	99.9 (30.5)	39.8 (19.3)
5	60	42.4 (20.1)	28.9 (23.0)	3.6 (2.4)	41.0 (18.9)

Table 2-11 – Comparison of average delays by flow

There are several trends readily observed in these results. First, the two optimization models tend to distribute delays in a markedly different fashion, as should be expected from the analytic results presented earlier. Qualitatively, FDP seems to match the trends exhibited by the final delay objective model, and RAP seems to match the trends exhibited by the total delay objective model.

It is important to evaluate each of these comparisons quantitatively as well. Table 2-12 represents the results of a series of *t*-tests at a 1% significance level used to compare the various flows. The numbers in this table represent the number of flows whose delays are statistically indistinguishable from one another for the pair of flows specified. Comparisons from a model to itself are made with respect to the overall mean delay for that model. Large numbers in this table represent a high degree of correspondence between the two models.

	Total	*Final*	*FDP*	*RAP*
Total	4	0	0	5
Final	-	2	1	0
FDP	-	-	0	0
RAP	-	-	-	4

Table 2-12 – Flow-wise comparisons of model results

On the diagonals of this table, the models are compared to their overall means. In each case, not all five flows have delays that are statistically indistinguishable from the mean. This suggests that each model treats some flows differently from others. The Total and RAP models perform best according to this metric, but in each case, Flow 1 is assigned a statistically significantly lower amount of delay.

These statistical tests also confirm the apparent trend that the RAP results mimic well those of the total delay model. While not the intention of examining this rule-based method, it is a valuable result worthy of further examination. Conversely, the results suggest that FDP does not do a very good job emulating the results of the final delay model. The magnitudes of the differences indicate that FDP is more aggressive in disadvantaging multi-resource flights than is the final delay model. Strategies to mitigate these differences are worthy of further consideration.

2.4.6 *Computational performance*

Another important issue highlighted in the first section of results was the differences in computational performance between each solution technique. The priority rule methods are excluded here because they are obviously extremely fast.

The important difference reflected in these results is that the optimization model with the total delay objective typically solves to optimality very quickly, while the final delay objective, which differs very minimally mathematically, takes considerable time to solve to provable optimality. While this difference is interesting, from a highly practical perspective, it is also important to quantify how quickly any good solution can be found by the final delay model, as it presents the greater computational challenge.

The metrics of computational performance are shown in Table 2-13. The data here reflect the amounts of time required to reach three milestones: finding the first integer feasible solution, finding the optimal solution, and proving that the best integer feasible solution is optimal. For these results, each model run time was capped at 36 hours, or when a gap from the best bound fell below 0.01%. The final delay model achieved only a 0.44% gap after this time.

Metric	**Model**	
	Total	*Final*
Time to first integer solution (seconds)	4.6	5.6
Time to find optimal solution (seconds)	79.3	>129600
Time to prove optimality (seconds)	79.3	>129600

Table 2-13 – Computational time comparison

An alternative method of visualizing the computational performance of these two formulation variants is depicted in Figure 2-21. In this, the time-varying optimality gap in both models is compared. On this time scale, the total model solves almost immediately, while the final model slowly decreases toward zero.

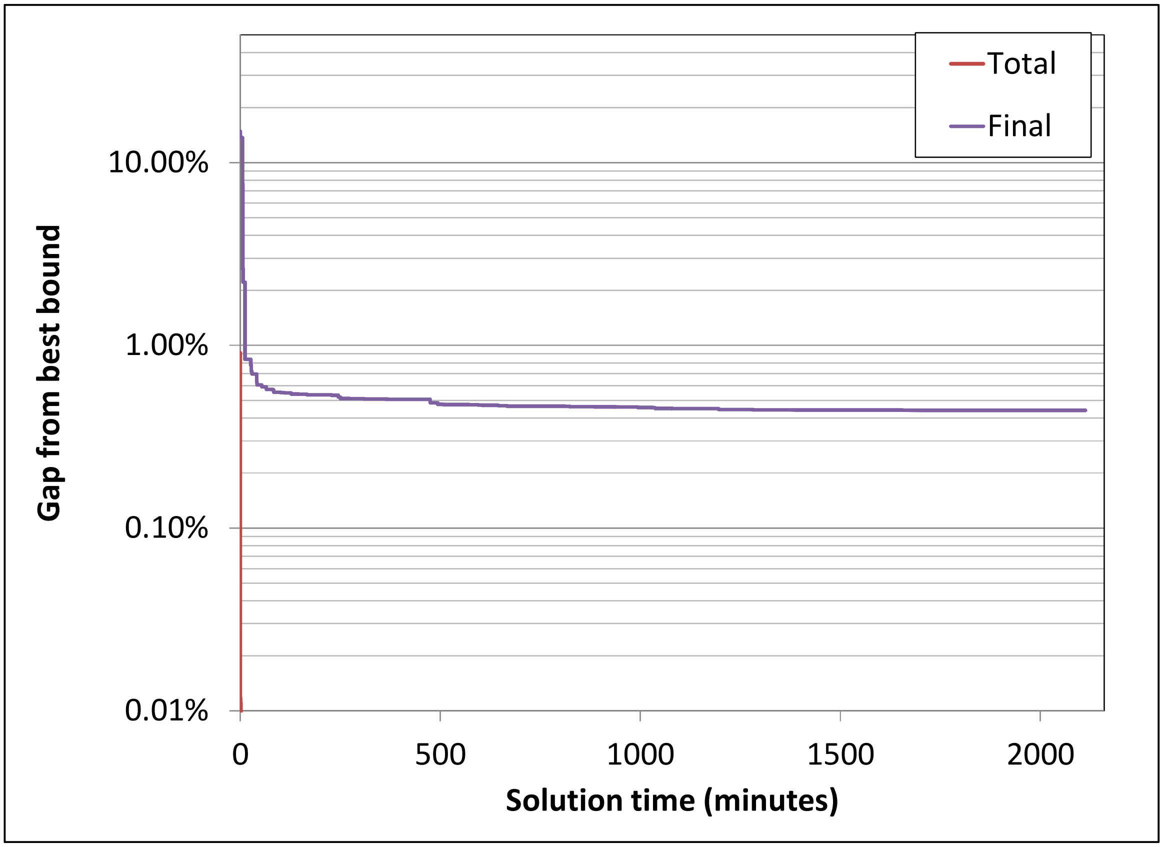


Figure 2-21 – Model computational performance

What is apparent from the previous table and figure is that both models rapidly find integer feasible solutions, although for the final delay model, these are not provably good. Solving other instances using these models has shown trends wherein the best integer feasible solution quickly achieves stability, but the bounds require many more minutes or hours to increase sufficiently to match this. In that case, the numbers typically reported for the gap are not representative of the difference from the optimal solution, but only from the poor bounds found thus far. This large deviation from the best known bounds represents the weakness of the linear programming relaxation. To this end, several techniques that seek to improve the strength of this formulation are explored in the next section.

2.4.7 ***Valid inequalities***

In this section, the value of the two valid inequalities presented earlier is explored. The intention of introducing these valid inequalities is to strengthen the optimization formulation. Given the relative ease with which the optimization model solves when using the total delay objective function, only the final delay objective function is considered in conjunction with these valid inequalities. A comparison of solution evolution is shown in Figure 2-22.

Performance is depicted for three variations: the base model with the final delay objective, the same with the backward-linking constraints added, and the base with the summation inequality added. Adding both valid inequalities precluded finding any integer feasible solutions within the allotted 15 minutes. These results suggest that the summation inequality improves model performance, as a lower bound is achieved most quickly of these three cases. It reached 1.0% first, after 8 minutes. The base model required nearly 15 minutes to reach that same milestone. However, although the performance is somewhat improved, the magnitude of the difference is not sufficient to unilaterally declare that this valid inequality improves performance significantly.

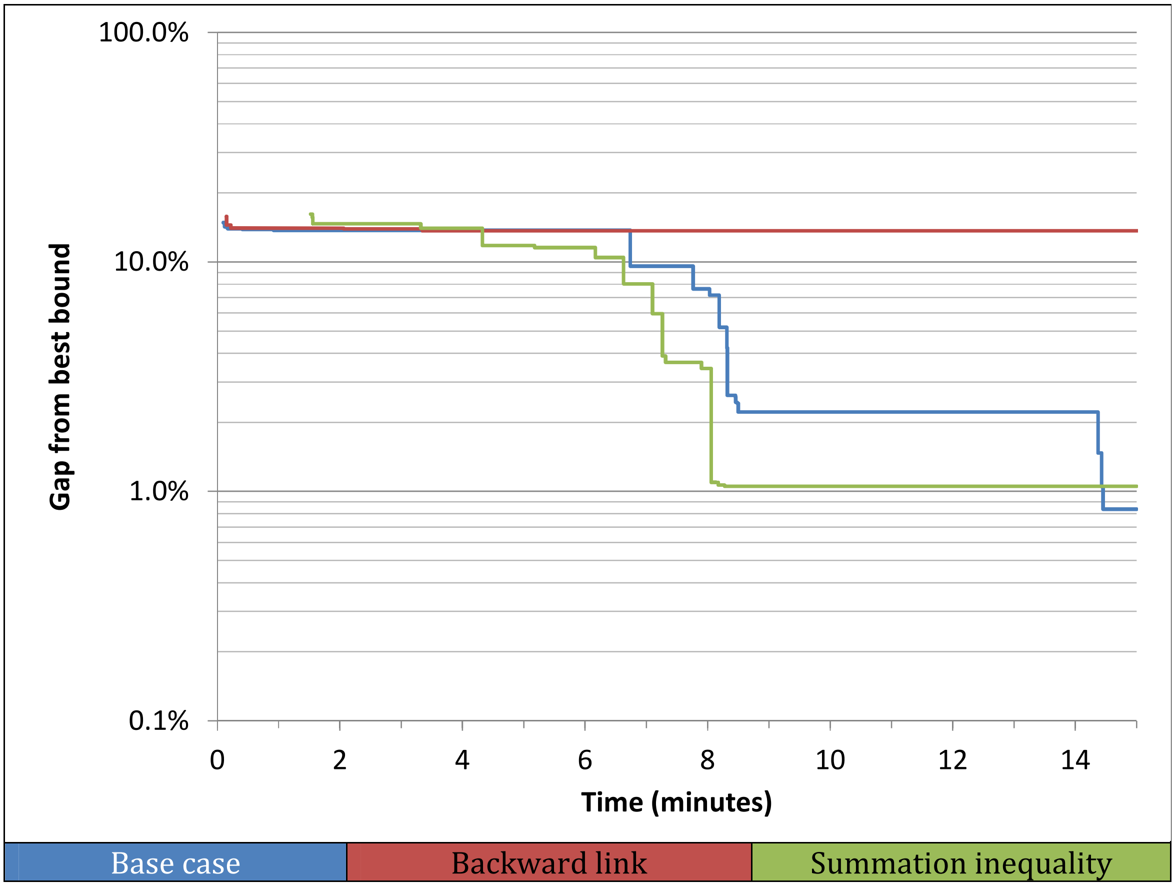


Figure 2-22 – Comparison of valid inequality performance

2.5 Conclusions

In this chapter, the problem of simultaneously allocating capacity at several resources was considered. Each flight may use a different subset of these congested resources, reflecting a situation outlined in detail in §1 and commonly encountered on busy days in the U.S. air traffic system.

The first approach proposed to address this problem was a linear optimization model. This model assigned flights to slots at each resource used by a flight to minimize delays. While doing this for a single resource in isolation is easy, the linking constraints that force compatible slot assignments between resources complicate the formulation. Two objective functions were considered for this formulation, and the implications of each were explored analytically. The total delay objective function encourages allocations consistent with the schedule used as a baseline, while the final delay objective prefers in general to prioritize flights based upon the number of resources they use.

However, recognizing some of the practical problems of employing optimization models, a rule-based approach was described. This was presented as a greedy heuristic that took as inputs the same data as the optimization models. The heuristic was also designed to take as input the explicit priority system intended. To this end, two priority schemes were evaluated – one that attempted to mimic the bias induced by final delay objective optimization model and one that attempted to model the operational implementation for coordination by prioritizing specific resources for allocation before others.

To demonstrate the efficacy of the two optimization variants, as well as the rule-based approach, a case study was undertaken. This confirmed the bias induced by the final delay objective function and aptly demonstrated that each approach performs as expected by minimizing the appropriate metrics. The rule-based approaches developed quality solutions very quickly. Their allocations matched fairly well with those from the optimization models.

However, computational evidence suggests that using the final delay objective function makes solving this problem through optimization much more difficult than using the total delay objective function. To this end, the computational performance was evaluated in greater detail, and it was found that good integer solutions are achieved rapidly. But, the bounds used in the branch and bound procedure are of low quality, thus giving the impression of low-quality integer solutions that would require long run times to alleviate.

The case study described in this chapter provides interesting material for analysis. However, although the final delay objective function is attractive for several reasons, implementing it or any rule-based approach based on it is impractical because of its impact on equity between users. To this end, the next chapter presents several approaches that extend those in this chapter to directly and explicitly control equity between users.

3. Equitable coordinated airspace capacity rationing

In the coordinated rationing scenario described in this paper, there is an inherent bias against flights using multiple resources when considering only flight arrival delays with the final delay objective function. This has been demonstrated empirically in the initial case study, as well as analytically in the previous argument. Although the resulting allocations, measured in objective terms, have desirable aggregate properties, the inequities introduced between user groups create an untenable political situation.

At first analysis of this problem, this bias represents a classic example of the unintended consequences of an optimization model – one class of users is disadvantaged to better the objective function. While the resulting solution is in mathematically optimal, practical concerns argue against it because it does not encompass other considerations that might render it optimal in a more global sense. In this section, several approaches for limiting or bounding the worst case performance of any individual user are proposed. Introducing such constraints will necessarily limit the resultant allocations to be at best no better than those derived from the base formulation, at least as measured in the aggregate.

This chapter is devoted to examining several mathematical methods to enforce or encourage fairness in the allocations developed by the optimization model shown in §2. First, several baselines from which deviations may be measured will be described. Then, several variations of maximum deviation constraints will be outlined that may be used to replace the assignment constraints in the base formulation from the previous chapter. Then, a cost-based approach is described that has various trades with respect to the constraint approach. Finally, the case study from the previous chapter is extended with several analyses to demonstrate the efficacy of explicitly including equity considerations in modeling this problem.

The constraint and cost-based approaches presented in this chapter all rely on having some baseline allocation of flights to slots that is reasonably accepted as being fair. This requires that some heuristic step precede any optimization or priority approach to determine this baseline allocation. The baseline to be used in this work as the fair allocation at each resource is the Ration By Schedule (RBS) allocation. It has long provided the basis for capacity rationing in the airspace system, and is accepted by user groups as fair because it is based on long term data (the published schedule) that cannot be manipulated for short-term gain.

The RBS procedure works by spacing out the nominal schedule to fit the newly defined lattice. Each flight is assigned, in schedule order, to the earliest slot that it can use. Denote the resulting RBS allocation slot time for flight *f* at resource *i* as a_f^i, and its position in that ordering as c_f^i. Recall that each flight has a scheduled arrival time α_f and scheduled arrival position γ_f. By construction, the following inequalities must hold:

$$a_f^i \geq \alpha_f^i \tag{3.1}$$

$$c_f^i \geq \gamma_f^i \tag{3.2}$$

3.1 Maximum deviation constraints

The first method examined for bounding the performance of an individual user is to place a hard constraint upon the maximum permissible deviation from the RBS allocation. This method provides a direct control upon the worst case performance of any individual user. This type of constrained assignment was first examined for managing air traffic by (Dear 1976). It has also been recently reexamined for real time systems to maximize runway throughput by reordering flights already en route and in the terminal area (Balakrishnan and Chandran 2010). In both of those cases, the application varies, but the intention remains the same:

constrain individual user performance by limiting deviations for the accepted fair sequence, that which is derived from the schedule order.

Two steps are included to implement this maximum deviation constraint: one is strictly necessary, while the other is convenient to decrease the size of the formulation. The first step is to change the set of possible assignments for each flight, while the second is to reduce the set of decision variables defined, as a consequence of the first step.

Equation (3.3) mirrors Assignment constraint I shown in (2.4), stating that each flight must be assigned to exactly one of some set of slots. In this section, several replacements for the set Q_f^i are proposed to meet the objectives of constraining the maximum deviation. In the original formulation, this assignment set was large, beginning at α_f^i and lasting until the end of the rationing program. In each of these cases, this set is smaller.

$$\sum_{s \in Q_f^i} x_{fs}^i = 1 \qquad \forall f \in F, i \in V_f \tag{3.3}$$

Following the same reasoning, when creating the decision variables $\{x_{fs}^i\}$, the new sets proposed here can be used to define existence. In each case, the number of variables created should be strictly less than in the base model, potentially improving computational performance.

Two alternative baselines are used for measuring the maximum deviation, drawing from the definition of the RBS allocation presented in the previous section: both the RBS allocation slot time a_f^i and the RBS allocation position c_f^i. While in many situations, there is a mapping between the results from these two baselines, dynamic capacity conditions will render these mappings invalid, making these two standards unequal in general. In using position shift metrics, time conditions are unnecessary, as in previous definitions, as the limits of the set are defined directly by the nominal position.

Although the mathematical differences between these two baselines are fairly subtle, the practical implications of this choice are somewhat more interesting. Specifying a maximum deviation as a length of time may be preferable because it provides a more tangible connection to the typical metrics of airline performance. Further, it may be simpler to specify any flight's worst case performance as, for example, 30 minutes. This type of standard would likely be useful across programs of varying capacities. Conversely, position may be desirable in this situation as a baseline because it provides an absolute measurement of flight performance irrespective of the resource capacities.

One complication in using time-based deviations is that they may not align precisely with the lattice of the slots to which the flight is assigned, as would position-based deviations. For example, if headways were 3 minutes under the capacity rationing, but the maximum deviation parameter was 20 minutes, then the seventh slot would not be fully encompassed in the range. In this model, however, the beginning time of each slot is used as the measurement point, and so the seventh slot would be included in this case.

Two types of deviations are admitted by these constraints. In the first type examined, only assignments later than the RBS allocation are permitted. In the second, both positive and negative deviations from the RBS allocation are permissible, but in either case, these are capped.

3.1.1 ***Negative deviations***

The first standard used for measuring deviations from a nominal assignment is that of negative deviation. In this case, a flight may only be moved later than its nominal assignment, but not earlier. This is depicted notionally in Figure 3-1, with the parameters ψ_U and ω_U controlling the maximum time or position shift, respectively. The time deviation parameter ψ_U is equivalent in use to that employed in constraining the Ration By Distance algorithm described in (Ball, Hoffman and Mukherjee 2009).

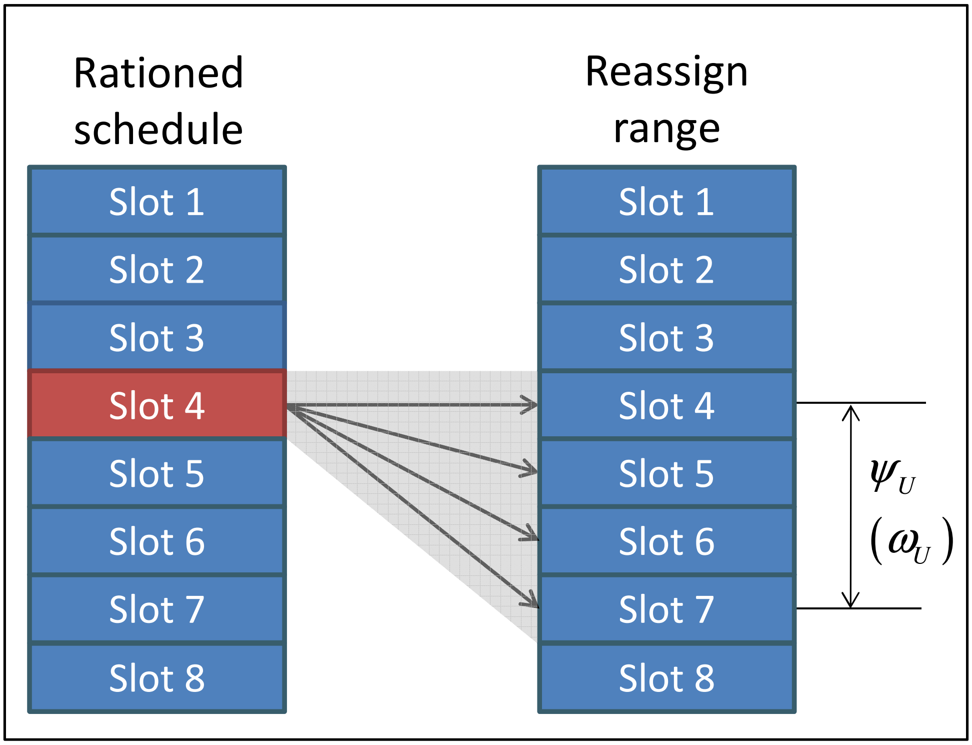


Figure 3-1 – Negative deviation constraint

Two new sets of slots to employ in the assignment constraints are shown here: (3.4) is used for time-based deviations, while (3.5) is for position shifts.

$$A_f^i = \left\{ s \in S^i : a_f^i \le \tau_s^i \le a_f^i + \psi_U \right\} \tag{3.4}$$

$$C_f^i = \left\{ c_f^i, \ldots, c_f^i + \omega_U \right\} \tag{3.5}$$

An interesting implication of these negative deviation constraints is that the solutions, while hopefully more equitable with respect to distribution of delays, may be poorer in the aggregate. In each case, the beginning of the set for feasible assignments begins with the RBS allocation, rather than the schedule. As a result, some flights will be disadvantaged with respect to what they would have received under the unconstrained allocation. The set of feasible solutions is smaller.

3.1.2 *Absolute deviations*

Similar to the concept proposed in the previous section, the deviation constraint here admits both earlier and later assignments, relative to the nominal RBS assignment. Earlier assignments are allowed up to ψ_L time units or ω_U positions before the RBS assignment. Later assignments are again constrained by ψ_U or ω_L. The nature of this constraint is depicted notionally in Figure 3-2. The set of slots permitted for a flight's assignment will be larger in this case than in the negative deviation case. To prevent conflicts with a flight's schedule, the lower bound of this time range is defined as the maximum of the deviation or the scheduled arrival time.

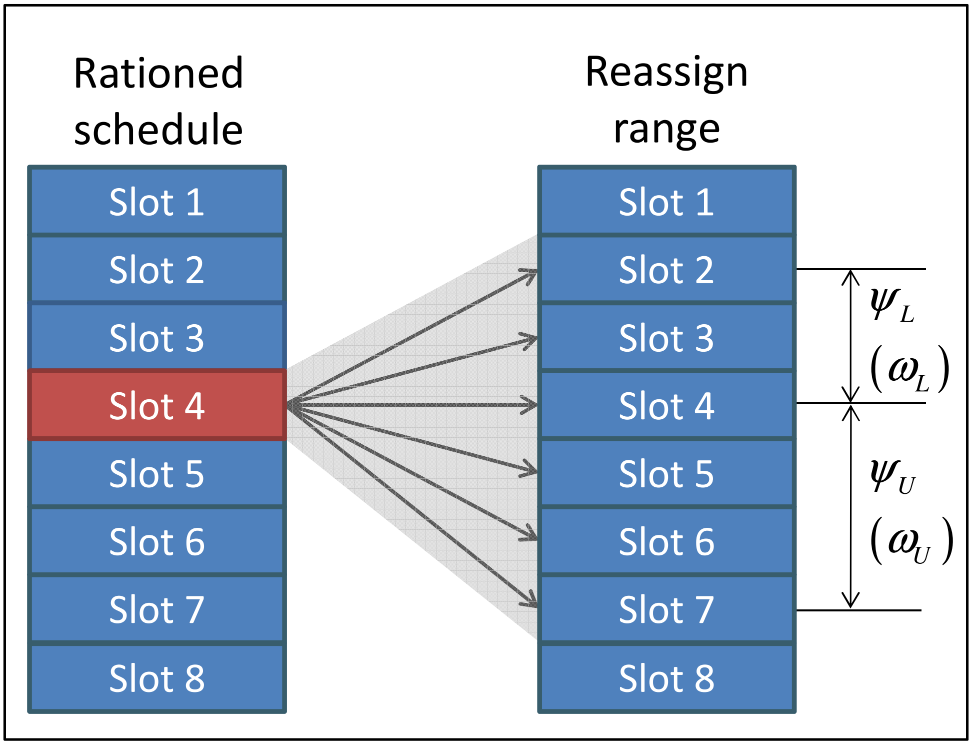


Figure 3-2 – Absolute deviation constraint

Two sets are again proposed to replace Q_f^i : (3.6) for time shifts and (3.7) for position shifts. One complication with these ranges is the necessity to set the lower bound at the maximum of the permissible deviation or the scheduled arrival time or position. This tightens the set of permitted slots, as by assumption, flights may not be assigned to a slot earlier than their schedule.

$$B_f^i = \left\{ s \in S^i : \max\left(\alpha_f^i, a_f^i - \psi_L\right) \le \tau_s^i \le a_f^i + \psi_U \right\} \tag{3.6}$$

$$D_f^i = \left\{ \max\left(\gamma_f, c_f^i - \omega_L\right), \ldots, c_f^i + \omega_U \right\} \tag{3.7}$$

Again, the solutions admitted by these constraints may be no better than those from the base formulations. However, in this case, it is certainly possible that the solutions would be very similar, as permitting earlier deviations will admit many of the changes that yield the quality solutions from the base model.

3.1.3 *Practical concerns*

There are many practical concerns related to employing some variety of these maximum deviation constraints. Because these constraints extend the formulation shown in the previous chapter, the practical considerations outlined there apply here as well. Although these maximum deviation constraints could be used with either of the proposed objective functions, there is little argument for employing them with the total delay objective, and a strong argument for doing so with the final delay objective. Because the total delay objective tends to maintain the flights schedule order, few large deviations will be observed. The final delay objective, however, provides the motivation for this chapter itself, because of the undesirable properties that its allocations exhibit.

The very first issue is which type of constraint should be employed: negative or absolute deviation, and time or position based. No specific choice is advocated here, but the positive and negative attributes of each choice will be addressed. On the first point, it seems that permitting both positive and negative deviations may decrease equity between users. Under the absolute deviation construct, two otherwise equivalent users may be shifted in opposite directions, magnifying any deviation needed to create feasible coordinated allocations. Permitting only negative deviations limits this inequity.

In either case, some users will be disadvantaged relative to what they believed they were due, and further, relative to what other users received. There is no easy solution available to placate these users. Although compliance according to law may be most desirable, it may be infeasible and may generate considerable ill will and discord. As a result, some system of credits compensating users for negative deviations may be employed. This is beyond the scope of the work proposed here, but has been explored elsewhere.

For either the absolute or negative deviation cases, some baseline must be employed against which deviations are measured. The two metrics proposed here were time and position. Time has the advantage of being conceptually easier, but may also complicate matters depending on the precise value chosen. A small value may limit extremely the set of slots to which a flight may be reassigned. Position is conceptually more complex, but provides more control in an absolute sense over how far a flight may stray from its nominal assignment.

Even if the above issues have been overcome, setting the precise value of the maximum deviation remains a difficult question. At any level, some users will object, however the value itself must be set large enough to provide any utility at all. If it is too small, there exists a possibility that no feasible coordinated slot allocation will even exist. This issue will be explored experimentally in the case study described later in this chapter.

3.2 Deviation costs

In this section, an alternative approach to limiting the performance of each individual flight is explored. In the previous constraints, the delay minimizing objective was employed with constraints on the maximum deviation from a nominal assignment. In this section, the objective itself will be changed to minimize deviations from the nominal assignment. Again, either the total or final delay construct could be employed with these objectives. However, given that the total delay objective function already produces allocations with some desirable properties, it may not be useful to consider in this context.

This approach is fundamentally different in that it does not explicitly place a bound on any flight's worst case performance. However, by measuring costs relative to the preferred ordering, the model is given a strong incentive to keep deviations from that preferred allocation small. However, because a hard constraint on the maximum deviation is not employed explicitly, there remains the possibility that a phenomenon such as that shown with the final delay objective could develop. This approach may allow longer delays (relative to RBS allocation) than are preferred, but this may allow allocations with other desirable properties. This may lead to undesirable outcomes, but needs to be explored more fully.

Of course, some variety of the maximum deviation constraints could be employed in concert with this new objective. That approach would provide the benefits of decreasing the formulation size, as was discussed in the previous section while allowing the model to determine the optimal deviations from the RBS allocation. This combined approach will be explored experimentally in the next section. As in the previous section, several methods of measuring deviations are considered, encompassing both positive and negative shifts, and using both time and position in the ordering as a baseline.

3.2.1 *Deviation objective functions*

In this section, several objective functions are defined to leverage the RBS allocation as a baseline for the coordinated allocation to be developed by the optimization model. The following measurements of deviation are included:

- Difference between assignment and baseline allocation: This approach assigns later deviations a positive cost and earlier deviations a negative cost. A minimization problem will clearly prefer the negative costs, and thus, may move some flights much earlier and others later to yield a net zero objective function. A superlinear function cannot be employed because of the earlier deviations permitted by this approach. This measurement is used in (3.8) and (3.11).
- Difference between assignment and baseline allocation with zero lower bound: This approach assigns positive costs to later deviations, and zero cost to earlier deviations. Thus, the incentive for moving flights earlier than their RBS allocation is removed, increasing equity between flights. However, no disincentive is provided against these earlier deviations. A superlinear function of delay length may again be employed to encourage multiple shorter delays because of the zero lower bound. This measurement is used in (3.9) and (3.12).
- Absolute difference between assignment and baseline allocation: The final metric considered assigns equal cost to both positive and negative deviations of equal magnitude. Thus, an equal disincentive is used to encourage that all flights receive allocations as close as possible to the baseline. This measurement is used in (3.10) and (3.13).

The above three deviation metrics are combined with two baselines against which the deviation can be measured. These are again the RBS allocation slot time – used in (3.8), (3.9), and (3.10) – and the RBS allocation position – used in (3.11), (3.12), and (3.13).

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \left(\tau_s^i - r_f^i \right) x_{fs}^i \tag{3.8}$$

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \max\left(0, \tau_s^i - r_f^i\right)^{1+\varepsilon} x_{fs}^i \tag{3.9}$$

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \left| \tau_s^i - r_f^i \right|^{1+\varepsilon} x_{fs}^i \tag{3.10}$$

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \left(s - c_f^i \right) x_{fs}^i \tag{3.11}$$

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \max\left(0, s - c_f^i\right)^{1+\varepsilon} x_{fs}^i \tag{3.12}$$

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \left| s - c_f^i \right|^{1+\varepsilon} x_{fs}^i \tag{3.13}$$

As described previously, each of these objective functions could be combined with the maximum deviation constraints from the previous section. In that case, the set Q_f^i in the summations should be replaced as appropriate. In addition, using one of the maximum deviation constraints may render the issue of deviation costs less than zero as moot, as the negative deviation constraints explicitly prevent early assignments.

3.2.2 *Practical considerations*

Using a cost-based approach to limiting the maximum deviation from a nominal allocation introduces several practical issues that bear addressing. As previously, these proposed objective functions represent an extension to the optimization formulation presented in the previous chapter. As a result, the caveats outlined there continue to apply.

However, one unique and pressing concern arises with this cost-based approach. Although this approach should tend to develop allocations close to the baseline, there is no guarantee that they must do so, as in the previous section. Consequently, undesirable allocations such as those in the previous chapter with a specific class of users disadvantaged remain feasible. As a result, the most prudent approach to employing these cost-based approaches may lie in conjunction with the maximum deviation constraints.

3.3 Multiobjective deviation costs

This section presents a generalization of the previous objective functions presented for the optimization approach to this problem. The objective presented in this section minimizes a weighted combination of flight arrival delays and deviations from the baseline allocation.

The general form of this multicriteria objective function is shown in (3.14). The summations follow the usual pattern established for the objectives presented previously considering delays at the time of each flight's arrival. The parameter σ is used to create a convex combination of the two objectives under consideration: Δ_{fs}^{i1} for the final delay for each flight, and Δ_{fs}^{i2} for the absolute deviations from the baseline RBS allocation.

$$\min z = \sum_{f \in F} \sum_{\substack{i \in V_f : \\ \left|N_f^i\right| > 0}} \sum_{s \in Q_f^i} \sigma\, \Delta_{fs}^{i1} + (1-\sigma)\Delta_{fs}^{i2} \tag{3.14}$$

Equations (3.15) and (3.16) reflect the formulae used to calculate the final delay and deviation from the RBS allocation for each flight, respectively.

$$\Delta_{fs}^{i1} = \left(\tau_s^i - \alpha_f^i\right)^{1+\varepsilon} x_{fs}^i \tag{3.15}$$

$$\Delta_{fs}^{i2} = \left|\tau_s^i - r_f^i\right|^{1+\varepsilon} x_{fs}^i \tag{3.16}$$

This objective function is a generalization. Taking σ = 1 yields the final delay objective, while taking σ = 0 gives the absolute RBS time deviation objective. Any value of σ between these extremes provides a weighted combination of these two.

The primary utility of this objective function is to examine various weighted combinations. Solving a single instance while varying this convexity parameter between 0 and 1 in small steps will allow for the identification of an efficient frontier representing the tradeoff made between efficiency (sum of final delays) and equity (sum of absolute deviations from RBS). This frontier will be examined in the case study that follows. Obviously this approach does not quickly lend itself to heuristic approximations, but may provide useful insight into this classic equity-efficiency tradeoff as it applies to this coordinated slot allocation problem.

3.4 Case study

To examine the efficacy of the equity-based methods proposed in this chapter, the case study first shown in Chapter 2 is extended. The same physical and temporal configuration is employed, with three congested resources. Several categories of results are examined, covering aggregate and detailed performance metrics for each variation proposed. Results are included for comparison from the previous case study where appropriate. Many of the analyses in this section are comparative. However, objective function values are not compared directly, as different objectives are employed in some cases. As a result, various properties of the allocations themselves are compared. The primary metrics used are the total delay, summed at each resource, and the final delay (arrival delay), summed for each flight upon arrival.

3.4.1 *Overall comparison*

The first set of result presented for the equity-based models are shown in Table 3-1. This is an extensive list comparing each of the equity-based models with both of the base models using the total and final delay metrics, measured in minutes. The two columns for each metric depict its value after two minutes of solution time, and at the optimal solution. For the models using maximum deviation constraints, shown in the second block of the table, the values used were 30 minutes or 15 positions, depending on which standard for measuring deviations was employed.

Model	**Total delay**		**Final delay**	
	2 minutes	*optimal*	*2 minutes*	*optimal*
Base – total	**14169**	**14169**	10703	10723
Base – final	14554	14394	10333	**8926**
Negative time deviation	15134	15134	11072	11072
Neg. position deviation	15123	15123	11072	11072
Absolute time deviation	14337	14394	9409	9406
Abs. position deviation	14374	14374	9843	9843
Time cost	14725	14853	**9177**	**8926**
Time cost with ZLB	14370	14370	10506	10506
Absolute time cost	14709	14709	10815	10820
Position cost	15405	14881	9494	**8926**
Position cost with ZLB	14361	14361	10490	10490
Absolute position cost	14700	14681	10804	10801

Table 3-1 – Comparison of equity-based models

There are many trends apparent from the many results shown in Table 3-1. First, one check of the validity of the results is made by confirming that the optimal allocations determined by the base models are no worse than any allocation developed by the cost-based or maximum deviation models. Next, the results demonstrate that both the time/position cost and absolute deviation models are able to produce quality allocations quickly. The negative deviation models are of lesser quality, but this should be expected because the set of feasible solutions must exclude any allocation in which flights are moved earlier in the sequence than they are due. Third, the results reported after 2 minutes and at optimality are equal for many of the models. This suggests that these models may identify an optimal solution quickly. Finally, it appears that the impact of the negative deviation constraints on the overall quality of the allocations is significant, given the difference in the amount of delays assigned versus the base models. Each of these issues, and others, will be explored in subsequent sections.

3.4.2 *Computational performance*

There is an important, but perhaps unintended, implication of employing the various equity-based methods shown in the previous section that is not entirely apparent. Some of these methods improve considerably the computational performance of the model in finding a good allocation.

Figure 3-3 compares the computational performance of the base model using the final delay objective against each of the maximum deviation constraint methods.

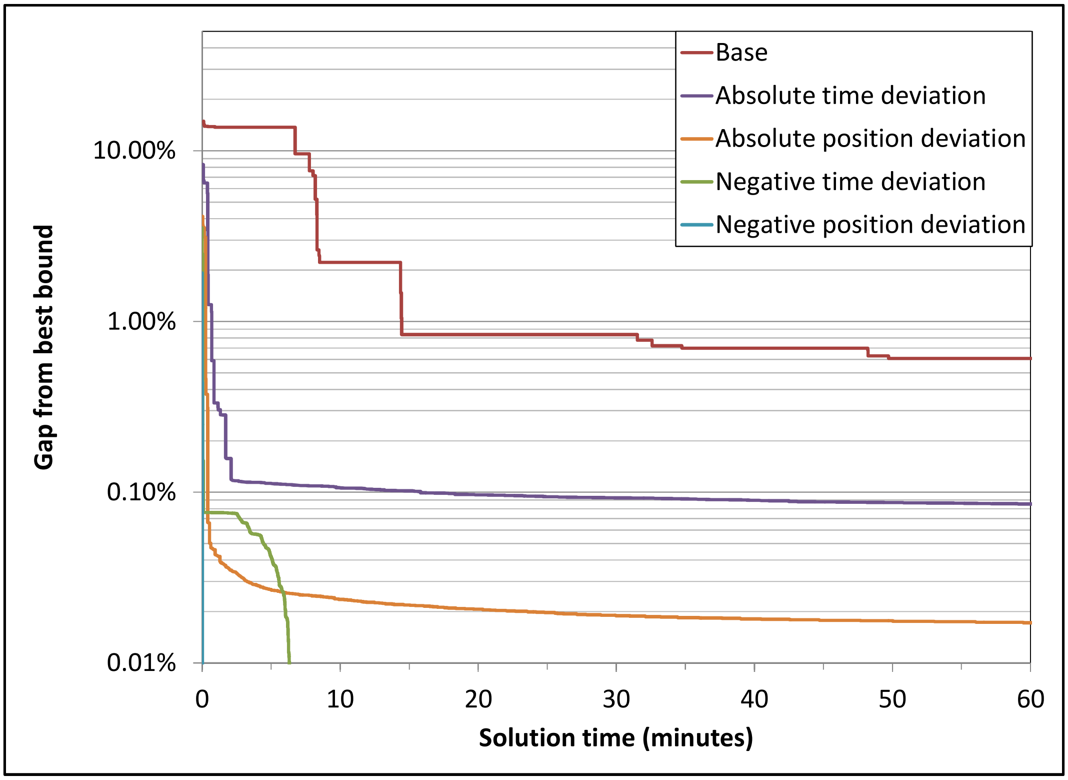


Figure 3-3 – Maximum deviation constraint computational performance

The metric used for the comparison is the so-called "optimality gap," which reflects the percentage difference between the best integer feasible solution and the best bound located thus far in the solution tree. Each of the maximum deviation constraint methods performs better during the entire solution process than the base model. The two negative deviation constraints perform better than the absolute ones, and the position-based models tend to solve more quickly than the time-based ones. Each of these trends indicates that these maximum deviation constraints have value not only in bounding the worst-case performance for any user, but also in improving the somewhat difficult computational problems presented by the first problem. Likely, the origin of this improvement is that the set of feasible allocations is smaller because the length of the time period over which each flight may be assigned is smaller.

Likewise, Figure 3-4 presents a comparison of the computational performance of each of the cost-based models against the base model.

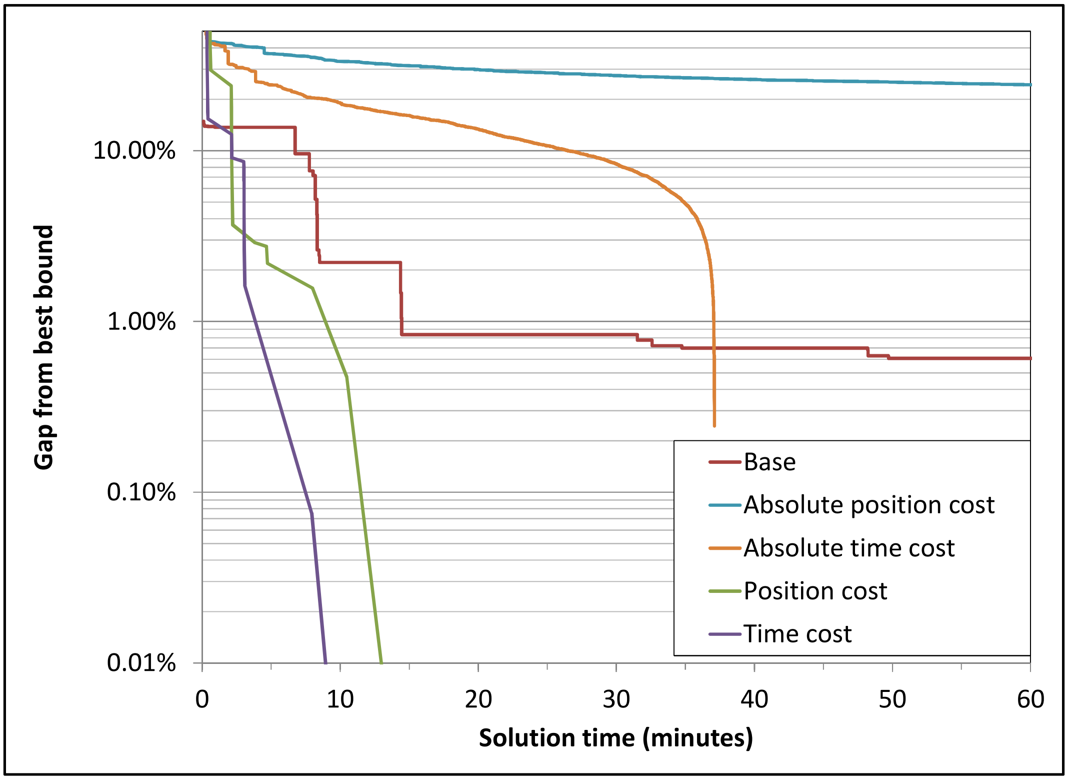


Figure 3-4 – Cost-based model computational performance

In this case, the results are somewhat less compelling, but nonetheless suggest that these alternate equity-based models may improve computational performance. From this figure, it appears that the time and position cost models performance quite well. This is an especially promising result, given that each produced an allocation with an amount of arrival delay equivalent to that from the base model. As before, the equity properties of this allocation are suspect, and will be explored next.

3.4.3 *Deviations from fair allocations*

An important metric of the performance of these models that seek to constrain the performance of any individual user is the actual deviations that each admits. A summary of the time-based deviations, measured in minutes, from the RBS allocation is shown in Table 3-2, while a summary of the position-based deviations is shown in Table 3-3. The data in these tables reflect the maximum positive (earlier), negative (later), and mean deviations from the RBS allocation.

Several trends are apparent in these tables. First, it is instructive to examine the second block of models – these represent the maximum deviation constraints. Three of these four models make use of the entire slack provided to them, excepting the negative time deviation. Remaining with the maximum deviation constraints, it is also interesting that the mean time and position deviations for the negative deviation models are greater than those for the absolute models. This is reasonable as the only feasible allocation for each flight is at or later than the RBS allocation.

Another trend is related to the size of the spread between minimum and maximum deviations. The largest spread is observed for the time/position cost-based models. These models permit any assignment later than a flight's scheduled time, as do the base models, but admit much greater deviations from nominal.

The absolute deviation cost-based models also develop allocations with interesting properties. The spread for each of these models is fairly small, although not the smallest in either case. As will be shown in the next section, these models also derive fairly equitable allocations.

Model	Deviation from RBS allocation time		
	Positive	*Negative*	*Mean*
Base – total	-12.0	13.2	0.00
Base – final	-52.9	81.3	0.58
Negative time deviation	0.0	21.0	2.63
Neg. position deviation	0.0	15.0	2.60
Absolute time deviation	-30.0	30.0	0.61
Abs. position deviation	-21.6	24.0	0.56
Time cost	-61.2	166.4	1.86
Time cost with ZLB	-31.0	18.0	0.55
Absolute time cost	-8.6	20	1.47
Position cost	-61.2	181.3	1.94
Position cost with ZLB	-31.0	18.0	0.52
Absolute position cost	-8.6	17.0	1.40

Table 3-2 – Time deviations from fair allocations

Model	Deviation from RBS allocation position		
	Positive	*Negative*	*Mean*
Base – total	-9	9	0.00
Base – final	-51	62	0.58
Negative time deviation	0	21	2.24
Neg. position deviation	0	15	2.20
Absolute time deviation	-30	27	0.61
Abs. position deviation	-15	15	0.56
Time cost	-51	139	1.76
Time cost with ZLB	-31	18	0.55
Absolute time cost	-6	20	1.44
Position cost	-51	150	1.87
Position cost with ZLB	-31	18	0.52
Absolute position cost	-6	17	1.37

Table 3-3 – Position deviations from fair allocations

3.4.4 *Comparison of treatment of various flows*

The next analysis considers the treatment of each flow, as defined earlier, by the above models. A comparison of these delays is shown in Table 3-4. The multi-resource flights are those in flows 2 and 4. The base model flow results demonstrated empirically the induced bias against multi-resource flights.

Model	Flow					Mean	Std. dev.
	1	*2*	*3*	*4*	*5*		
Base – total	32.0	40.1	44.2	38.1	42.4	39.4	18.6
Base – final	1.1	59.2	28.6	55.3	28.9	32.8	28.3
Negative time deviation	33.5	43.6	43.2	41.3	42.4	40.7	19.0
Neg. position deviation	33.5	43.5	43.3	41.4	42.4	40.7	19.0
Absolute time deviation	9.3	51.9	34.6	50.0	33.1	34.6	23.4
Abs. position deviation	16.9	48.0	37.7	46.0	36.1	36.2	21.4
Time cost	1.1	65.0	23.9	61.4	24.1	32.8	43.8
Time cost with ZLB	28.3	42.5	42.2	39.7	41.1	38.6	18.6
Absolute time cost	33.7	42.5	42.3	39.7	41.1	39.8	18.6
Position cost	1.1	67.3	22.0	59.4	25.7	32.8	44.3
Position cost with ZLB	28.0	42.5	42.2	39.7	41.1	38.6	18.6
Absolute position cost	33.4	42.4	42.3	39.6	41.2	39.7	18.5

Table 3-4 – Summary of average arrival delays by flow

There are many interesting trends evident in this table. First, the negative deviation models exhibit higher overall delays because they are not permitted to allocate flights earlier than they are due. However, the resulting allocations are more equitable, and achieving this was the target. Next, the absolute deviation constraints seem to mimic the trends observed for the base model with the final delay objective. This should also be expected, as these models will strive to develop allocations as close to that, while staying within the bounds set for them. For the cost models, it is clear that the simple time and position cost functions yield results even more extreme, with multi-resource flights being more disadvantaged, than those from the base model. The cost models with ZLB and the absolute cost models each perform well, and several actually dominate the base model with the total delay objective function, which may be considered in this context as the standard for coordinated equity.

3.4.5 *Effects of varying maximum deviations*

An important issue from political, practical, and computational perspectives is the effect of varying the maximum deviation parameters in the four constraints proposed in this chapter. If the maximum deviation, specified either in time or position terms, is sufficiently small, then the model may struggle to find a feasible solution. Conversely, if the range is excessively large, the model essentially defaults to the base model proposed in the previous chapter. From a practical and computational perspective, it is important to know what maximum deviation value admits a quality solution, but is solvable quickly. Finally, political concerns dictate, to a certain degree, the range of maximum deviation values that may be used. Clearly, choosing the best value is a difficult proposition. In this subsection, both the absolute and negative position deviation constraints are examined. These are used in place of the time-based constraint because of their superior computational performance, which is an asset when solving the many instances for this analysis.

The analysis in this section is built upon solving many instances of the base model with the maximum position deviation at various values. For the absolute case, symmetric deviations were used, and in each case the solution process was stopped when the optimality gap reached 1.0%. The negative deviation case is, in general, easier to solve, and so an optimality gap of 0.1% was employed.

The first set of results pertains to the relationship between the maximum deviation admitted and the quantity of delay assigned, as depicted in Figure 3-5. As expected, the smaller the maximum deviation permitted within in the absolute model, the greater the amount of arrival delay assigned. Using any maximum deviation less than 5 yielded infeasible problems. The rate of change of this curve is interesting in itself. At a maximum deviation of seven positions, the slope changes significantly. Below that point, the allocations quickly become much worse, while above it, the rate of change is much slower and quite steady.

For the negative deviation model, the results are significantly different. No feasible solutions are admitted below a maximum deviation of 12 positions, but there is essentially no variation in average delays over this range. This suggests that increasing the maximum has little marginal effect on the optimal allocation.

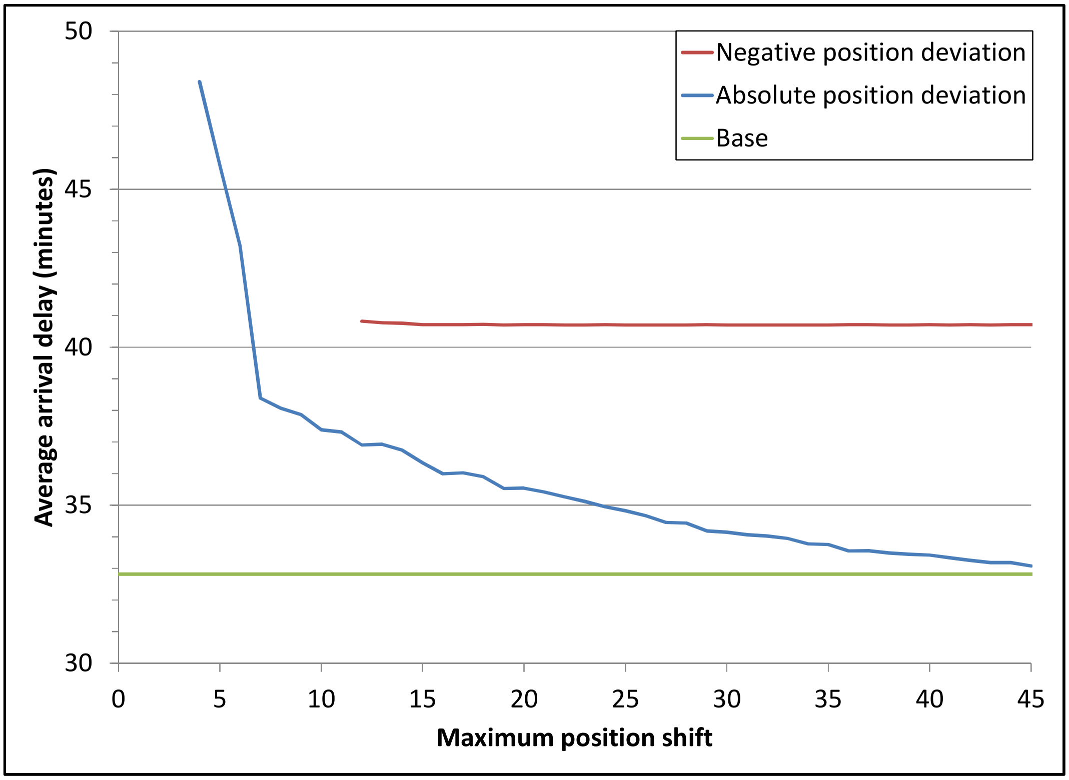


Figure 3-5 – Variation in assigned delay with maximum position shift

The curve goes only to a maximum deviation of 45 units – it would need to proceed to a negative deviation of 51 and a positive deviation of 62 for the absolute model to achieve the optimal solution found by the final delay base model.

The results in the first figure indicate that increasing the maximum permitted deviation in each model yields different results. For absolute model tends to improve with an increasing range, while the negative model changes little. This behavior is confirmed in Figure 3-6, which shows the maximum permitted deviation against the maximum assigned deviation. The absolute model uses, in every case, all the slack made available to it. In constraints, the negative model uses less slack than it could beyond a maximum of 20 positions.

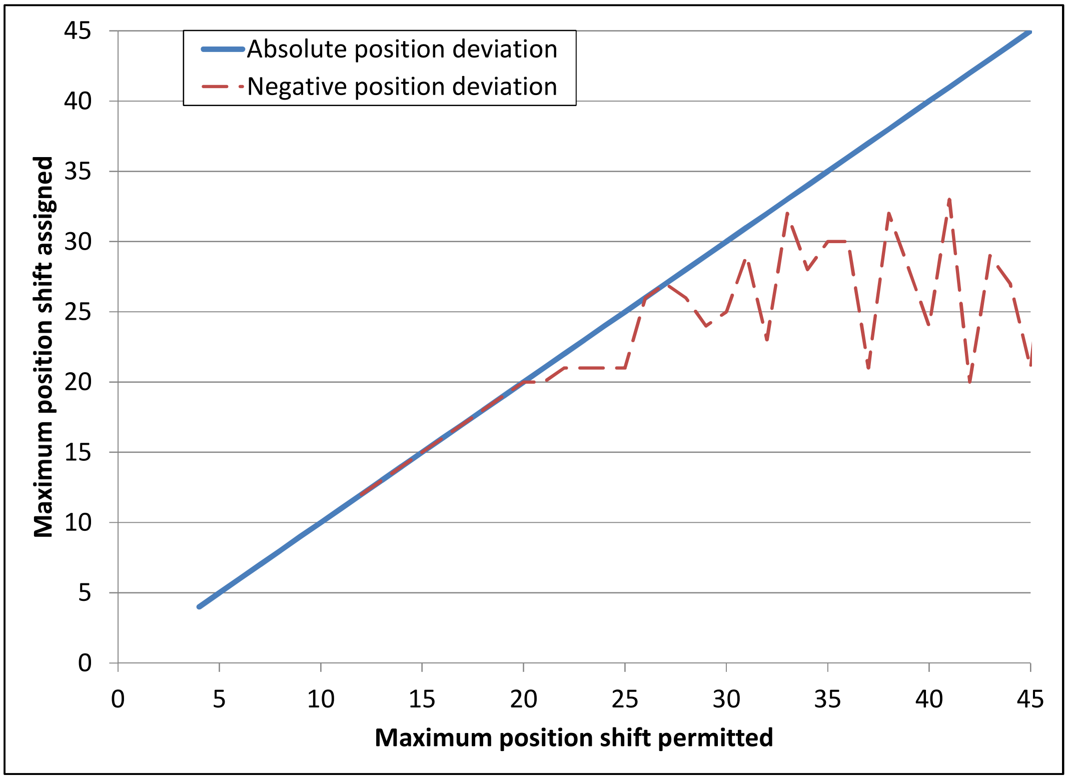


Figure 3-6 – Maximum assigned position shift

The above results begin to suggest that increasing the maximum deviation beyond a certain point will yield no marginal benefit for the negative model, and will yield marginally decreasing benefits for the absolute model. However, the size of this maximum deviation permitted also has an effect on the solution time required for each instance. Thus, it is prudent to choose an optimal value based not only upon its delay-minimizing benefits, but also according to its solution time. Figure 3-7 depicts the variation in solution time with the permissible maximum deviation. The obvious trend in this figure is that, for the absolute model, solution time increases directly with the maximum position shift. Despite the noise in this curve, the trend is apparent. This phenomenon likely occurs because the larger deviations admit a greater number of feasible solutions. Thus, it takes more time to search these feasible integer solutions and find the optimal one. Again, the trend in the negative deviation model is that increasing the maximum position shift induces little variation.

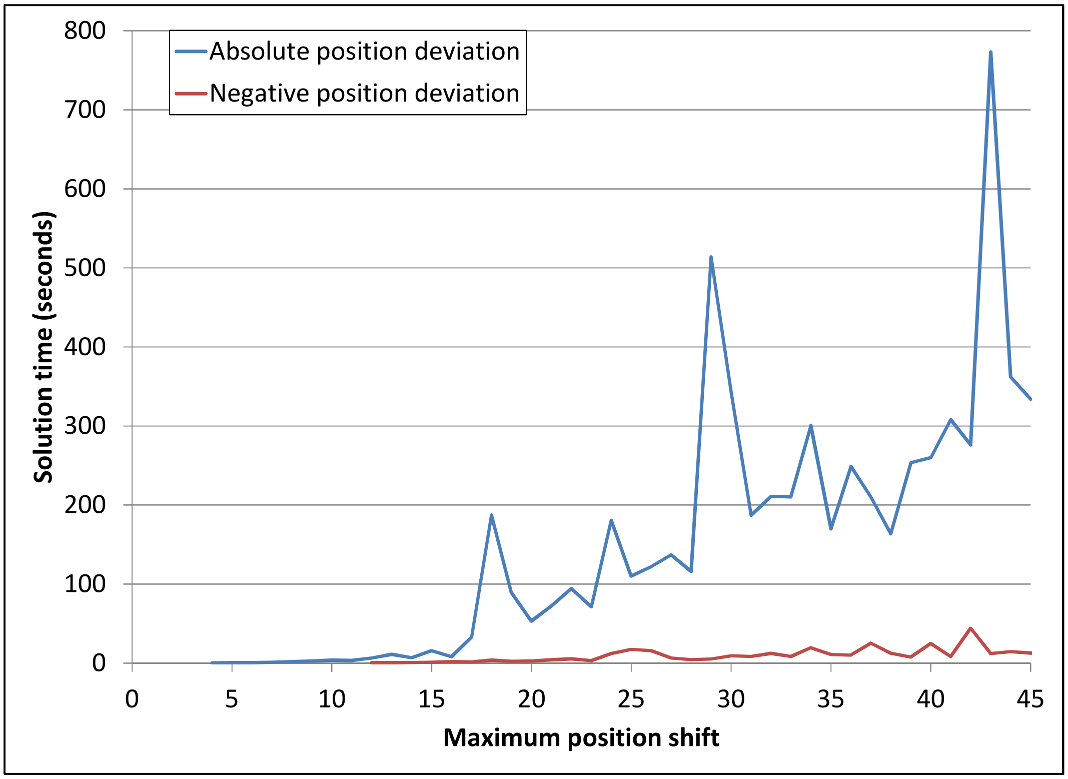


Figure 3-7 – Variation in solution time with maximum position shift

Although average delay assigned for a given maximum position shift is an interesting metric, it covers only one half of the efficiency-equity trade of which this chapter is concerned. In Figure 3-8, the standard deviation of assigned arrival delays is shown as a function of maximum permitted position shift. The same trend as observed above continues – the relationship is positive for the absolute model, and constant for the negative model. Again, the absolute results are consistent with expectations because the greater the maximum permitted deviation, the closer the allocation will be to that derived from the base unconstrained model. For the negative deviation constraint, these results again suggest that there is little marginal benefit to increasing the maximum deviation beyond the minimum feasible value.

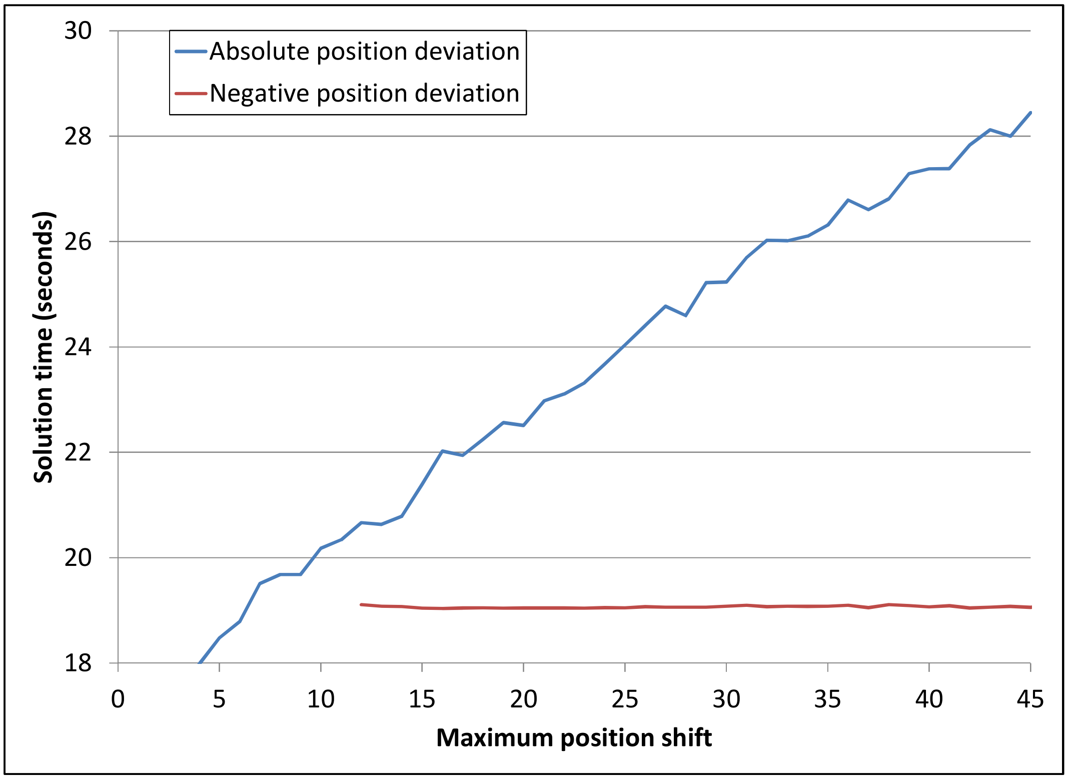


Figure 3-8 – Variation in standard deviation with maximum position shift

Finally, the results of the efficiency and equity analysis are consolidated into a single figure. In Figure 3-9, each pair of mean arrival delay and standard deviation of arrival delay are plotted against one another. For the absolute model, this forms a fairly smooth curve characterizing the tradeoff between efficiency and some notion of fairness. The results for the negative deviation constraint model however are reduced to a single point in the figure. This confirms finally that there is no benefit for efficiency or equity in increasing the maximum deviation beyond the minimum value to ensure feasibility. Given the rapid solution times for this model, and the valuable property of constraining worst-case performance of any user, applying it iteratively to find the minimum value may be a very effective strategy for developing coordinated capacity allocations.

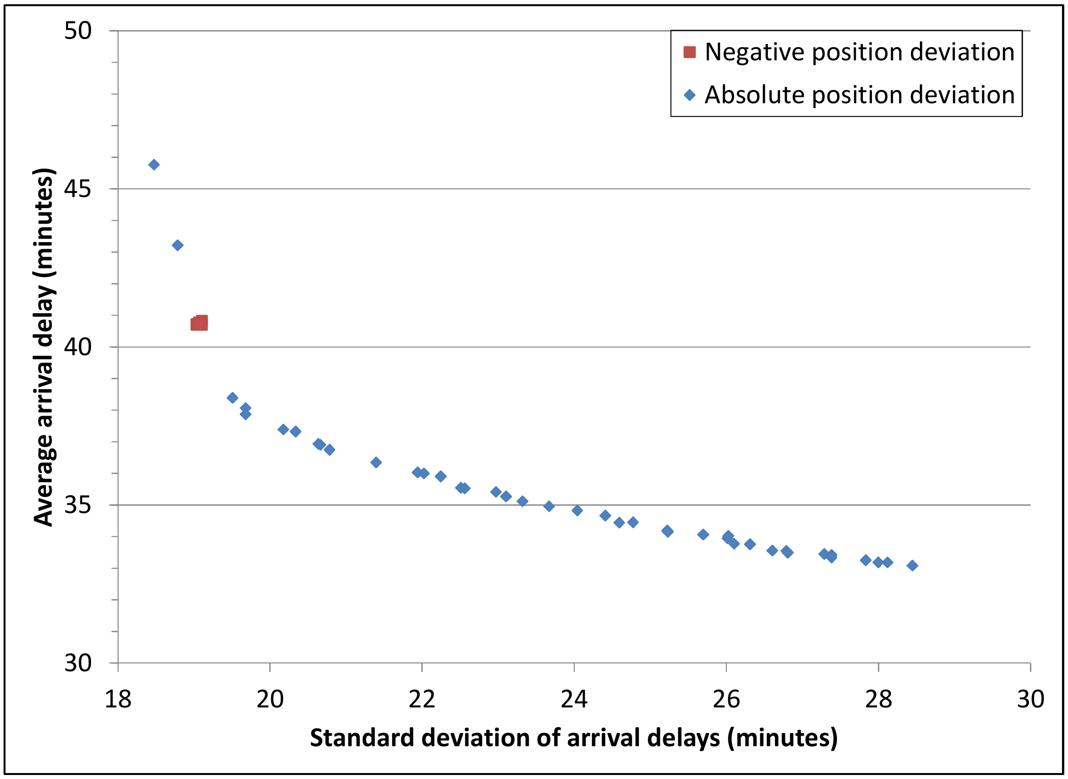


Figure 3-9 – Efficiency/equity frontier

Several sections have examined the properties of allocations derived from the maximum deviation constraints and the cost-based approaches separately. In the next section, these two concepts are combined.

3.4.6 *Combining equity methods*

Building on the results shown in the previous section, the approach proposed here aims to combine the cost-based and maximum deviation methodologies. The intention here is two-fold. First, high quality solutions may be obtained very quickly by combining the most efficient equity-based methods. Second, the resulting solutions will retain the desirable properties of each approach.

The first summary of these results is shown in Table 3-5. The aggregate performance of each model is compared to the base model and to each of its constituent models. It is clear that each of these combined models assumes the properties of the applicable constraint. This is reasonable of course, given that the constraint defines the set of feasible allocations, whereas the objective function only chooses the best of these. However, speed gains in determining these solutions are negligible, rendering this approach of questionable utility.

In Table 3-6, the performance of each model in allocating delays to each flow is examined. Similar trends are observed, in that the combined models assume the properties of the constrained model.

For this instance, the approach of combining constraints and cost-based equity methods is of little utility, in that it does not yield improved allocations and improves solution time only marginally. These combinations may be of greater utility for larger instances in which solution time may become a more critical factor.

Model	**Total delay**		**Final delay**	
	2 minutes	*optimal*	*2 minutes*	*optimal*
Base – final	14554	14382*	10333	**8926***
Negative time deviation	15134	15134	11072	11072
Time cost	14725	14853	**9177**	**8926**
Time cost & negative time deviation	15132	15132	11072	11072
Neg. position deviation	15123	15123	11072	11072
Position cost	15405	14881	9494	**8926**
Pos. cost & neg. pos. dev.	15150	15150	11104	11104

Table 3-5 – Average delay comparison for combination method

Model	**Flow**					**Mean**	**Std. dev.**
	1	*2*	*3*	*4*	*5*		
Base – final	1.1	59.2	28.6	55.3	28.9	32.8	28.3
Negative time deviation	33.5	43.6	43.2	41.3	42.4	40.7	19.0
Time cost	1.1	65.0	23.9	61.4	24.1	32.8	43.8
Time cost & negative time deviation	33.5	43.3	43.5	41.9	42.0	40.7	19.3
Neg. position deviation	33.5	43.5	43.3	41.4	42.4	40.7	19.0
Position cost	1.1	67.3	22.0	59.4	25.7	32.8	44.3
Pos. cost & neg. pos. dev.	34.0	43.4	43.4	41.4	42.3	40.8	19.3

Table 3-6 – Summary of average delays by flow

3.4.7 *Multiobjective optimization*

The final analysis presented here considers the multiobjective model proposed earlier in this chapter. The intention of this analysis is to examine the frontier of solutions formed by convex combinations of the final delay and the absolute time deviation objectives.

Figure 3-10 depicts the tradeoffs in efficiency (average arrival delay) with variations in the convexity parameter. The curve in this figure forms an interesting shape with the exception of the points at 0.60 and 0.65. This break is simply due to the poor computational performance (large optimality gap) for those two cases. This figure suggests that the optimal convexity value for minimizing delays (maximizing efficiency) lies at 1.0, or setting the objective function simply equal to the total delay model.

Again, however, this figure represents only efficiency. Figure 3-11 shows the variations in the equity metric (standard deviation of assigned arrival delays) against the convexity parameter. Noise is again observed in the same range. However, it is clear that minimizing the variance in assigned delays (maximizing equity) requires that the convexity parameter be as small as possible. This corresponds to the absolute RBS time shift objective.

Clearly the above two conclusions contradict one another. It may be most efficient to select a combination near the center of this range to apply equal weights to the two objective functions. Alternatively, Figure 3-12 presents the efficiency-equity frontier, comparing the mean delays with their respective standard deviations. The noise in this figure prevents the identification of a concrete inflection points. The results of this analysis do not provide any compelling arguments for employing this objective function as a means to determine the optimal tradeoff between efficient and equitable allocations.

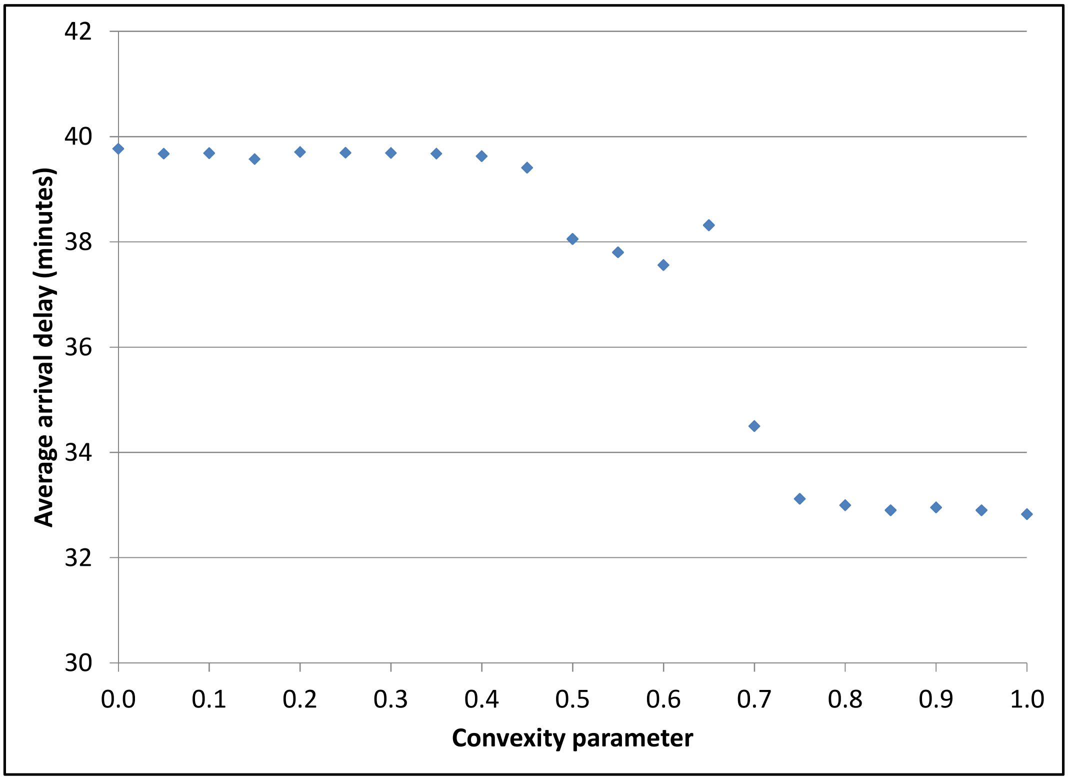


Figure 3-10 – Variation in average arrival delay with convexity parameter

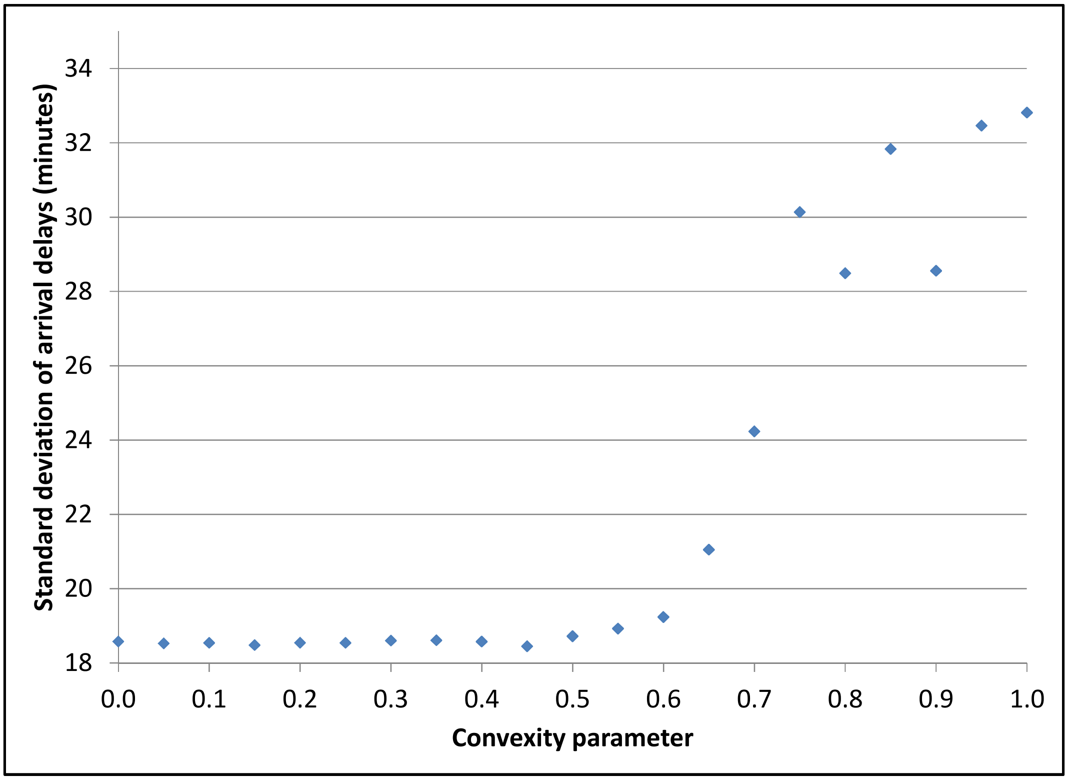


Figure 3-11 – Variation in standard deviation with convexity parameter

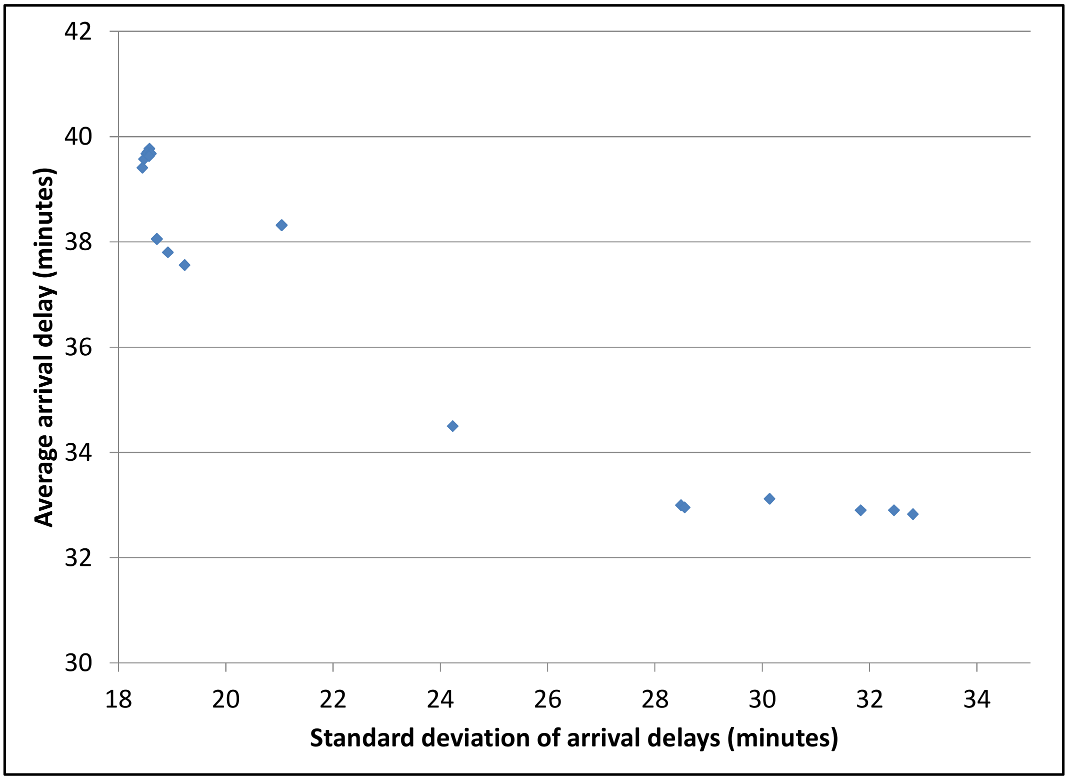


Figure 3-12 – Efficiency/equity frontier

3.5 Conclusions

In this chapter, numerous variations on the base optimization model presented in §1 were presented. These aim to explicitly control equity in making coordinated slot allocations. This chapter necessarily began with a discussion outlining what is the equitable allocation from which deviations should be measured. Based on this, constraints limiting either the absolute or negative deviations from the baseline were proposed. Then, several cost-based approaches were outlined.

Two baselines were used for measuring deviations, both derived from the initial RBS allocation. The time slot to which a flight was assigned under RBS, as well as the position in the ordering, were both considered. The constraint approaches varied on whether both positive and negative deviations were permitted. The cost-based approaches considered the simple difference between the assigned slot and the RBS allocation, that difference with a zero lower bound, and the absolute value of that difference.

Many results were presented in this section, covering the efficiency and equity properties of each of the variations proposed. The constrained models developed high quality allocations, each in alignment with its expected properties. The cost-based models produced interesting results, but none provided a compelling candidate for explicitly regulating the equity of coordinated capacity allocations.

The effects of varying the maximum deviation parameter for the constraints proposed were examined. These produced the interesting result that, for the negative deviation constraints, increasing the maximum beyond the minimum value required for feasibility produced little marginal benefit. This is a very valuable result, as it indicates that every flight's worst-case performance can be tightly bounded, while simultaneously developing a quality allocation. As an additional benefit, these constraints strengthen the optimization formulation and yield shorter solution times.

The equity-based variations on the base optimization model presented in this chapter represent useful compromises in developing an optimization-based approach to deterministic coordinated capacity rationing. However, none of the models presented in this, nor the previous chapter, address the possible uncertainties present in this system. In the next chapter, these uncertainties will be examined, and a new model formulated that explicitly considers them.

4. Stochastic coordinated airspace capacity rationing

Air traffic flow management decisions are highly sensitive to capacity disruptions lasting several hours or more at a time. Airports, regions of airspace, and specific control points throughout the airspace system may be affected. Disruptions may be caused by a variety of phenomena including primarily weather systems, but also equipment outages, security concerns, and military operations.

Many models have been proposed and employed in making ATFM decisions. The primary influencing factor in these models and their resulting decisions is the balance between demand and capacity. These models are necessarily sensitive to the uncertainty included in the predictions for available capacity under disrupted conditions. Considerable unpredictability is induced in the ATFM system by both the demand and capacity processes, and some researchers have attempted to explicitly include these uncertainties within several ATFM decision-making models, including rationing access to a single resource (Richetta and Odoni 1993) (Mukherjee and Hansen 2007), and to a limited degree, the multi-airport problem (Vranas, Bertsimas and Odoni 1994).

These capacity disruptions may be characterized, at a simple level, by several parameters: onset, severity, and duration. Each of these parameters is, however, difficult to predict for many types of disruptions. Weather systems in particular are inherently stochastic because of the myriad factors that drive their evolution. Further, the impact of the weather on the available capacity at any airspace resource is difficult to quantify, adding an additional level to the uncertainty involved in planning around capacity disruptions.

In this chapter, the problem of developing coordinated slot allocations at a sequence of connected resources is considered under stochastic assumptions about capacity, using the simple model described above. First, the impact of capacity uncertainty is quantified for allocations made using the deterministic optimization model introduced in §2. Because this analysis demonstrates that capacity uncertainty has a significant impact, an integer programming model is developed that explicitly includes information about the stochastic evolution of resource capacities. After the formulation is described, a case study is shown that builds on those for the deterministic formulations.

4.1 Assessing the impact of stochastic capacity variation

This research quantifies the effects of random variation on models considering only deterministic forecasts of capacity. While the base model proposed in §2 is employed, it is representative of the general class of ATFM decision-making models available. The several parameters characterizing capacity disruptions are considered, with random variations added to each, across many random demand scenarios. This Monte Carlo simulation technique allows for generalized, statistically meaningful conclusions about the impact of stochastic variations in capacity disruptions for airspace resources.

To improve planning models and procedures, it is important to understand in quantifiable terms what impact these uncertainties have on the system, and on decision-making. Thus, to assess the impact of random variations or mispredictions in available resource capacity, the deterministic optimization model for coordinated air traffic flow management decisions introduced in §2 is employed. Particularly with an optimization model such as this, it is important to understand whether results are a function of the model structure, large trends in the input data, or smaller, seemingly random variations in parameters, as well as the sensitivity of the model to each of these variations.

In the next subsection, the several modes of capacity variation are described in greater detail and the experimental design used in constructing the simulations is outlined. Then, results are reported evaluating the impact of these variations on the coordinated ATFM model.

4.1.1 *Modes of capacity variation*

To examine model sensitivity to variations in capacity data, several modes of capacity variation are considered. These are included as they represent three parameters of a simple model of capacity variation. Under this construct, a resource (e.g., an airport or airspace sector) has some nominal capacity at which it is operating. Then, due to a disruption, typically the movement of a weather system into the area, the capacity is reduced for some amount of time. The resource capacity remains at this reduced value until the disruption clears. At that time, the resource capacity returns to nominal. This simple model is considered for both airports and congested airspace regions. This model of capacity is employed because its simple structure allows for systematic analysis of the effect of capacity variation. Limiting the degrees of freedom in this manner simplifies the computational analysis needed to generate meaningful results. Simple models of capacity of this nature have been employed before in ATFM model, typically focusing on disruption end time, as in (Ganji, et al. 2009) and (Cook and Wood 2009).

The three modes of capacity variation considered are shown in Figure 4-1. The first mode, shown in Figure 4-1a, is the length of the disrupted period. This is the amount of time during which the nominal capacity is lowered to the reduced value, and the primary period over which access rights must be strictly controlled. This value is measured in units of time.

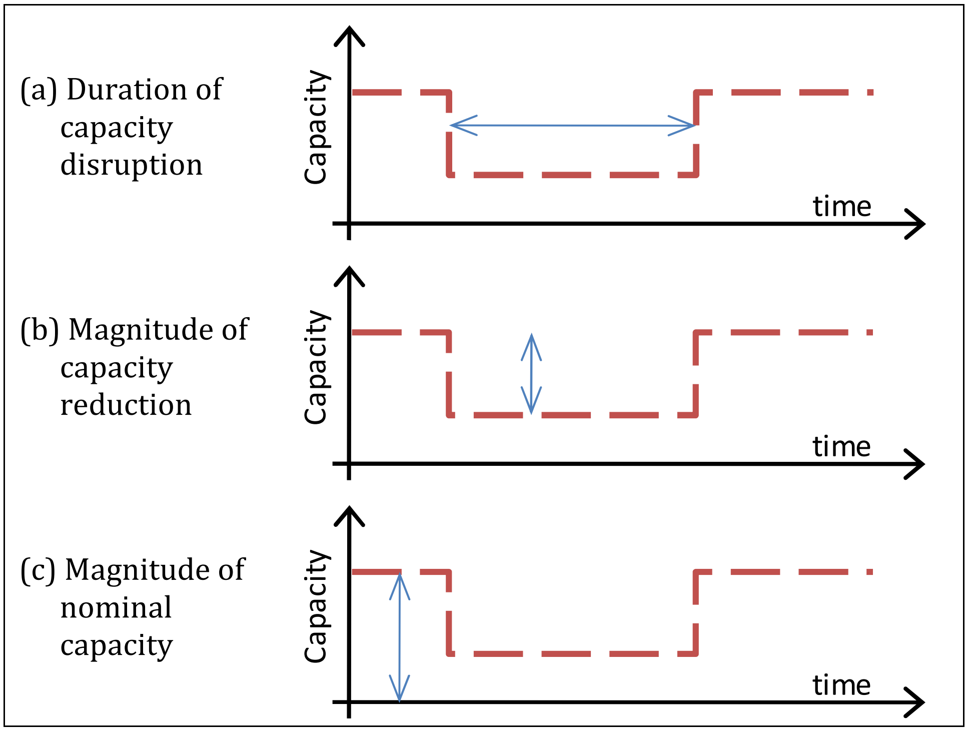

(a) Duration of capacity disruption
Capacity
time
(b) Magnitude of capacity reduction
Capacity
time
(c) Magnitude of nominal capacity
Capacity
time

Figure 4-1 – Modes of capacity variation.

The second mode of capacity variation, shown in Figure 4-1b, is the decrease from nominal capacity. This represents the amount by which capacity is reduced during the disruption. This may be measured as a relative change (percentage) or as an absolute change (flights per hour).

The final mode considered in this study is shown in Figure 4-1c. It does not deal directly with the severity of the capacity disruption, but rather with the nominal capacity level against which the disruption is measured. It is measured in units of throughput or capacity, typically flights per hour.

4.1.2 *Simulation model structure*

To examine the impact of these systematic variations in capacity, a Monte Carlo simulation model is employed. Each iteration of this simulation model will consist of solving the previously described optimization model with a random variation on demand and capacity profiles for each resource considered. After many such iterations, the results are aggregated to identify useful trends.

Under such an approach, however, it is important to carefully define which parameters may vary randomly, and by what means this may occur. In this case, there are many candidates: three modes of capacity variation (individually or in concert), schedule parameters including numbers of flights and their paths, and the number of airports and airspace regions considered. The primary objective of the research is to examine the result of varying capacity parameters, but only very limited conclusions might be drawn, without also varying the other parameters in some fashion.

To this end, the following procedure is employed:

1. Identify system configuration: number of airports and airspace regions, and baseline schedule parameters
2. Identify parameter(s) of capacity variation to be examined
3. Generate random variation on baseline schedule
4. Generate capacity profiles spanning possible range of variations (e.g., ±40%)
5. Solve model with generated demand and each capacity profile
6. Repeat steps 2-5 until statistically valid conclusions are reached

Following these steps will allow for generating conclusions conditioned on some baseline system configuration and mode of capacity variation, rather than on a specific demand profile or generic capacity description. This process works because the parameters identified in the simple model of capacity variation allow for systemic variation. Rather than having to generate many random permutations of the capacity profile, this process simply sweeps across a range of reasonable variations to generate results. In addition, solving the model for each of several demand scenarios for the range of capacity variations helps to eliminate dependencies on unique, but random, features of a single demand scenario.

Thus, this analysis is in fact a sensitivity analysis against systematic variations in capacity. Randomness is introduced to the schedules considered to provide greater validity for the results, as they will not rely simply on the structure of a single demand instance. It is important to recognize that both demand and capacity vary in each instance, with demand doing so randomly, according to the methodology proposed in the case study in §2, and the capacity varying systematically around some nominal value.

4.1.3 *Numerical results*

The model described in §2 with the total delay objective function was tested according to its sensitivity to variations in the three capacity modes described above. The spatial configurations for these experiments follow nominally from that shown in Figure 1-10, but are defined precisely in Table 4-1.

Characteristic	**Spatial configuration**		
	(1)	*(2)*	*(3)*
Number of airports	1	2	3
Number of airspace regions	1	1	1
Inter-resource travel time (minutes)	60	60	60

Table 4-1 – Spatial configurations for base scenarios

Flight paths were generated randomly for each flight, with flights using only one resource with probability 60%, and two resources with probability 40%. Each resource had a nominal schedule of 60 flights per hour; however as part of the randomness introduced in the simulation process this was reduced. At airports, flights were eliminated from this nominal schedule with probability 20%, while at the airspace regions, flights were eliminated with 50% probability. This larger fraction was used to allow space for flights destined for affected airports.

Capacity parameters were varied systemically for each of the three modes. The nominal length considered for each disruption was three hours, but this was varied in small steps up to ±40% for each demand scenario. Likewise, the fraction of capacity lost during the disruption was nominally 40%, but this was varied in small steps ±40% from the nominal value. Finally, the nominal initial capacity was 60 flights per hour, but this varied in small steps ±40%.

The primary results of the simulation are shown in Table 4-2. The results here are measured in terms of elasticity of average delay per flight with respect to unit changes in the capacity mode being considered. Thus, higher values reflect greater sensitivity to a particular parameter. Each elasticity value is the median of all those calculated over each range of capacity variation, each for ten demand scenarios. In nearly all cases, the quantity of delay assigned is elastic to unit changes in capacity. The effect decreases with increasing spatial complexity, but nonetheless suggests that this model is sensitive to perturbations in the precise values of capacity input. That many demand scenarios were evaluated for each combination of spatial configuration and capacity mode lends credibility to the results that they do not represent one lucky permutation.

Spatial configuration	**Capacity Mode**		
	(a)	*(b)*	*(c)*
(1)	1.82	2.27	4.56
(2)	1.26	1.54	3.23
(3)	0.78	1.03	2.36

Table 4-2 – Median elasticity of average delays to capacity variations

Another method to examine the variations across configurations and modes of capacity variation is shown in Figure 4-2. In this case, the various lines represent the average delay per flight, as it varies across the capacity scenario considered. Each line represents the average of all the demand scenarios considered for that particular experiment. Thus, for a given configuration, the lines for each of the capacity modes should cross at the 0% deviation mark, indicating that each scenario was being solved for the nominal conditions.

Further, the varying shapes of the curves depicted in Figure 4-2 provide useful information. For variations in capacity modes *(a)* and *(c)*, the curves appear to be concave up, indicating that variations in capacity lead to marginally increasing variations in average delay assigned. Capacity mode *(b)* exhibits a different behavior, of marginally decreasing. It is interesting to note that despite being marginally decreasing in this domain, changes in average delay as shown in Table 4-2 are still elastic with respect to variations in capacity.

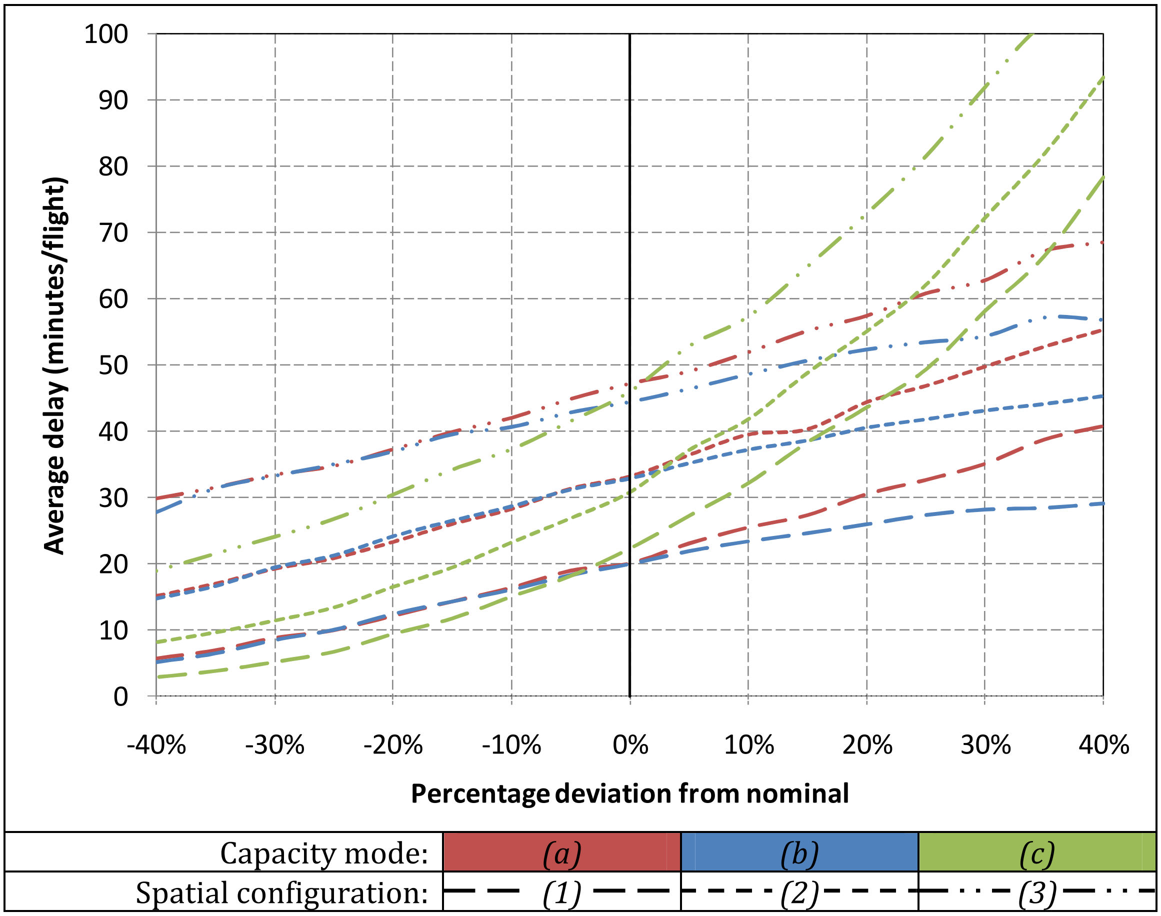


Figure 4-2 – Relationship of average delay and capacity deviation

A summary of the computational performance of these results is shown in Table 4-3, with sample sizes and median solution times (in seconds) for each combination of spatial configuration and capacity mode examined. For each capacity mode and spatial configuration, 17 capacity scenarios were used, each of which with 10 demand scenarios. This represents at most 1530 integer programming models to solve; however only 1475 were used as they converged to within 5% of optimality within 30 minutes. A total of 546 hours of computational time were required to solve these models. This was reduced considerably by solving several cases simultaneously on an eight core, 16GB Xeon computer system with Xpress 2008b.

	Capacity mode	*(a)*	*(b)*	*(c)*
Spatial configuration	**Number of cases used**	499	497	479
(1)	509	4	7	5
(2)	497	76	380	133
(3)	469	2230	2651	2397

Table 4-3 – Sample size and median solution times

The experiments described in this section provide an interesting example of the interplay between simulation and optimization techniques. The methods described here provide an empirical method to quantify the sensitivity of a particular optimization model to various input data. Systematically quantifying this relationship for any optimization model provides a greater understanding of their often opaque nature. In this case, a better understanding of this relationship provides excellent motivation for the natural evolution of the model to explicitly include stochastic, scenario-based data.

4.2 Stochastic capacity descriptions in ATFM

To frame the model proposed in this chapter, it is important to understand the nature of stochastic descriptions of aviation capacity that are available. The varieties useful in this context, and for the previous research described, are based upon scenarios that are realized at some time. Two varieties considered here may be described separately as disjoint, or tree-based.

An example of a disjoint stochastic description is shown in Figure 4-3, derived from data from (Liu, Hansen and Mukherjee 2008). In this example, there are four different capacity profiles that may be realized on a given day. They are specified as disjoint - it becomes known immediately which scenario will occur.

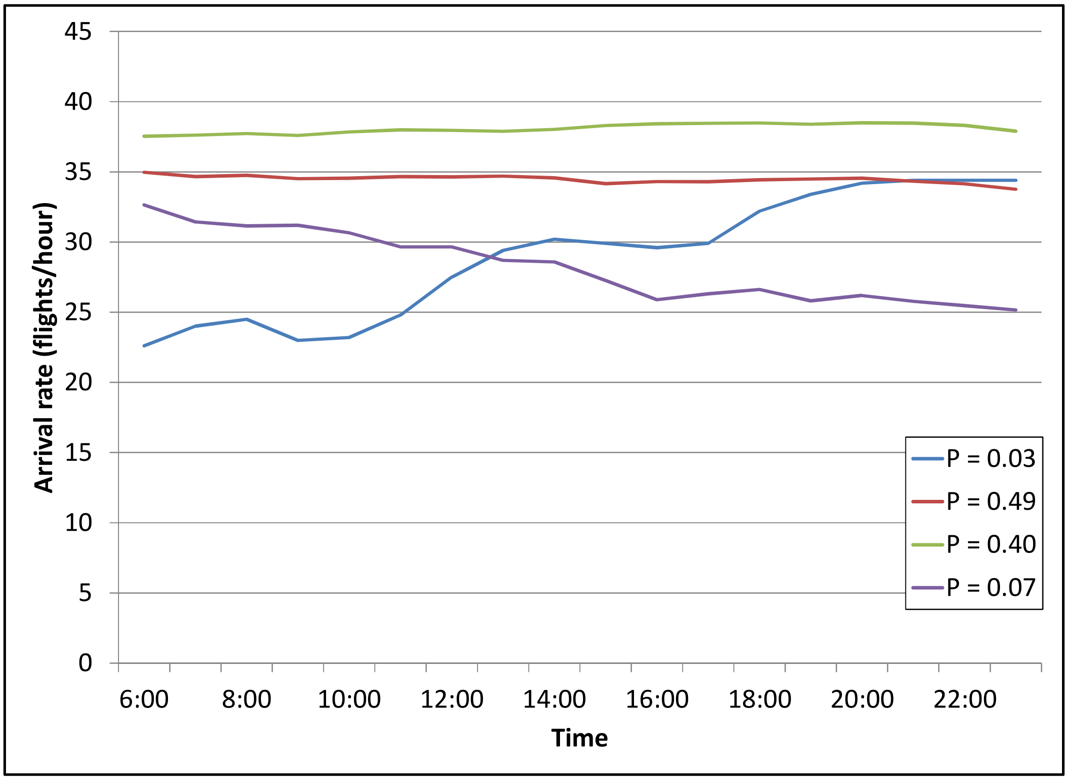


Figure 4-3 – Discrete capacity scenarios

While this approach is compatible with the model presented here, it is likely more suited to longer term planning efforts such as (Churchill 2007) because the value of the recourse actions suggested by the optimization model becomes dubious.

An expanded model in (Buxi and Hansen 2010) attempts to develop these scenarios on a time scale more useful to ATFM processes, but the utility of these is not considered here because alternative specifications that are more detailed about the dynamics of capacity evolution are available.

An alternative method to consider stochastic capacity is through the use of tree-based descriptions. Under this paradigm, a tree branches into different capacity realizations at specific time epochs. This construct is generalizeable to an n-stage decision process and was employed to allocate capacity at a single resource in (Mukherjee and Hansen 2007). However, this construct accommodates simpler models of capacity evolution. For example, a commonly explored issue in characterizing stochastic capacity evolution is to examine the variation in end times for a capacity disruption. In this case, at each time epoch, the issue is whether the weather has yet cleared, and conditions returned to nominal. This may be modeled using a simple binary tree, as depicted in Figure 4-4. In this example, weather clearance has some probability at each half hour interval, and must clear by 12:00.

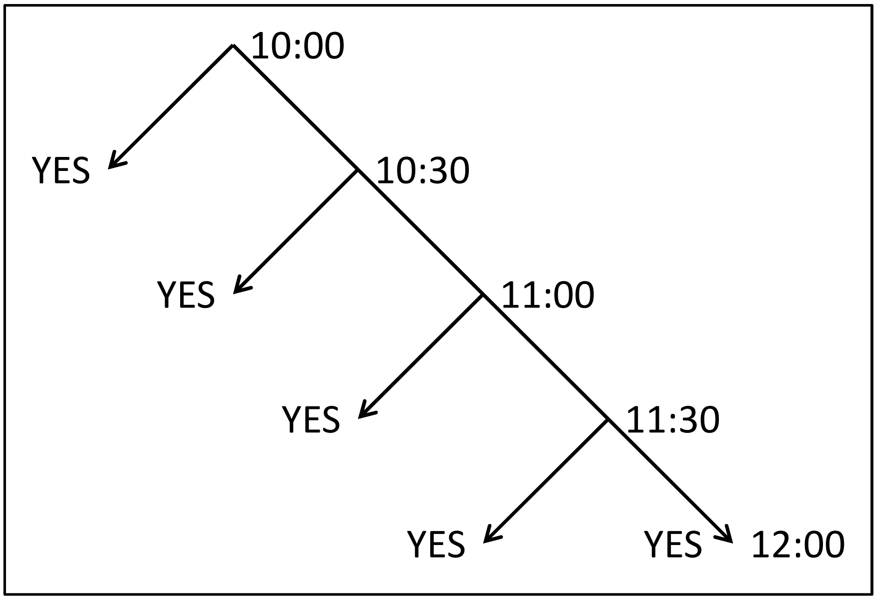


Figure 4-4 – Simple binary scenario tree

To implement this decision structure, data such as that shown in Figure 4-5 are used. In this example, the time spacing between decision points is only 30 minutes, but the cumulative distribution function of the possible clearance times is depicted as well as each interval.

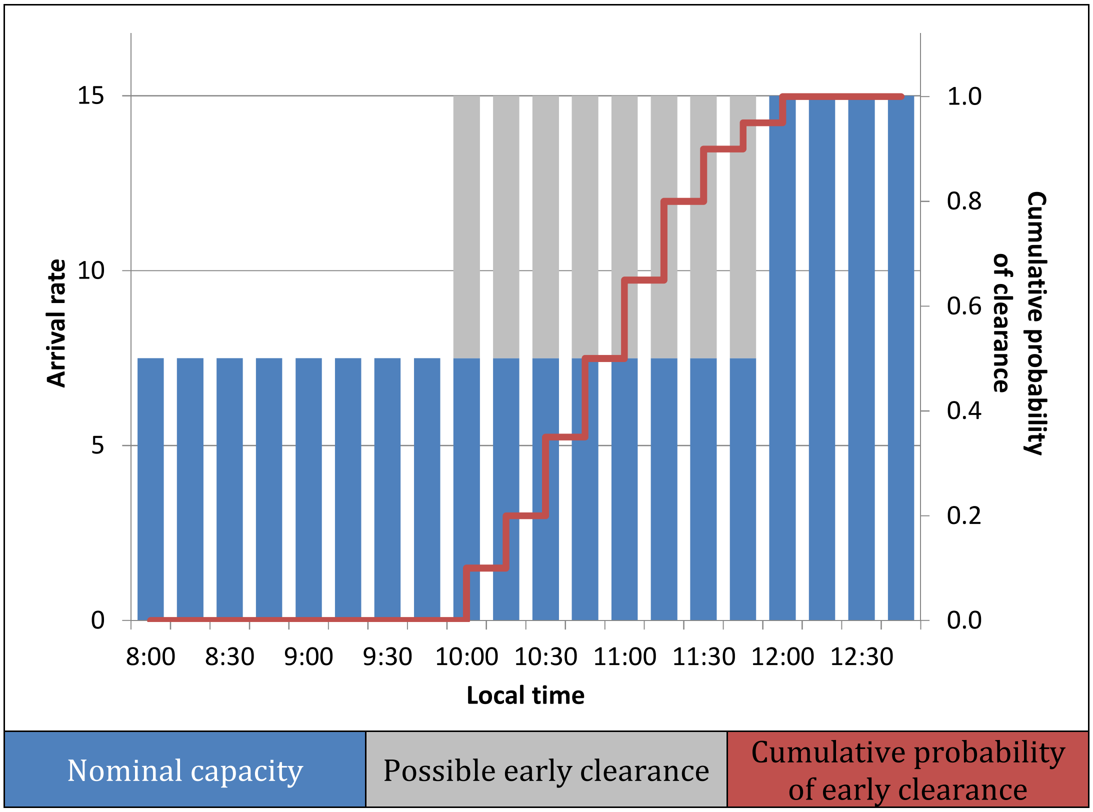


Figure 4-5 – Two stage capacity scenarios

In this example, the resource is experiencing decreased capacity, but it becomes possible starting at 10:00 that conditions will clear and capacity will return. Each subsequent time epoch is assigned a probability of this clearance, with the cumulative function of these probabilities shown in the figure. By 12:00, capacity will definitely have returned to nominal, and thus the CDF reaches 1.0 at that time. This type of capacity construct is employed in (Ganji, et al. 2009), (Cook and Wood 2009), and (Glover and Ball 2010) among others.

Building on the need for coordination in ATFM, and the available stochastic resource capacity descriptions, the next section introduces a stochastic integer program to explicitly model these coordination effects under uncertainty.

4.3 Optimization formulation

Evidence from the first section of this chapter suggests that the deterministic optimization models considered in this thesis are sensitive to variations in capacity. Explicitly considering stochastic capacity evolution should then, in some contexts, provide benefits to coordinated slot allocations. In this section, a stochastic integer programming extending the base model described earlier is introduced.

Specifically, this model assigns flights to arrival times at each of a sequence of resources that the flight encounters between origin and destination. A resource may be an airport, some congested portion of airspace, or any other airspace resource of finite capacity. Only resources expected to be congested are considered. Structurally, each resource is considered as an assignment problem, but linking constraints are included to insure that each flight using multiple resources receives compatible slot assignments.

Uncertainty in capacity outcomes is incorporated through the use of a two-stage stochastic formulation, wherein both an initial plan and conditional plans for each outcome are developed. The model presented here considers the simple model of capacity disruptions presented in the previous section. The formulation and the case study shown here are designed to consider only responding to uncertainty regarding the end time of the capacity disruption. A discussion for relaxing this assumption to consider more general capacity models is included, but a complete reformulation is not presented.

The two stage formulation allows for only a single change in the capacity conditions experienced, allowing for tractable models to be developed. For each scenario, a plan corresponding to that capacity outcome is developed. These recourse actions represent those that should be undertaken, given the prescribed set of initial decisions, to optimally respond to the changes realized in capacity. The nature of the recourse actions specified will of course depend on the capacity changes for that scenario, but may include dispatching flights currently on the ground, further delaying flights on the ground, or assigning airborne holding to flights already en route.

The model presented here represents a greater degree of control than is currently exerted by system operators today, but is not incompatible with the principle of collaborative decision making (CDM) that have been so widely adopted in ATFM. The decisions developed during the first stage of the model represent the initial assignments that would be made, but there is no reason that individual airlines or users, with their collections of slots at each resource, could not perform their own swaps or trades to meet their internal objectives. While this may detract from the system-level objectives espoused by this model, they do represent the ability of users to optimize their operations within the construct provided. The second stage decisions prescribed here do not represent decisions that must be implemented, but rather, are the optimal decisions, given the appropriate set up provided by the first stage decisions.

The same model inputs and notation are used from the presentation of the base model, with the addition of data to represent the stochastic capacity outcomes. This is included here through a set of scenarios Q, as described in the previous section. For each q in Q, there is an associated probability of occurrence (p^q) and realization time (t^q). The realization time represents the time at which it becomes certain that a given scenario will occur, but may not necessarily represent the time at which the capacity change occurs. It is used to model the ability to forecast capacity improvements, typically because of expected weather clearance. In addition, each scenario has associated slot times τ_s^{qi}. The slot times in τ_s^{qi} are equivalent to those in τ_s^i before t^q, but begin to vary then or at some later time. The initial conditions are included as a scenario in the set *Q* to allow some probability to be assigned thereto. For the simple model of capacity employed here, each scenario will represent a different potential early end time for the capacity disruption.

In the remainder of this section, the mathematical structure and properties of this formulation are described, including decision variables, constraints, objective functions, and computational properties.

4.3.1 *Decision variables*

Two related sets of binary decision variables are employed in this formulation. The first set is used to define the initial decisions, the second to define the conditional decisions for each scenario.

The first set of decision variables, $\{x_{fs}^i\}$, defines the initial plan – a value of one indicates that flight *f* was assigned to slot *s* at airspace resource *i*. These variables are defined for each flight, and for each slot that it may feasibly use – namely those with a beginning time equal to or later than the flight's scheduled arrival time, as shown in (5.1).

$$x_{fs}^i \quad binary \qquad \forall f \in F, i \in V_f, s \in Q_f^i \tag{5.1}$$

The second set of decision variables $\{y_{fs}^q\}$ are similar to the first, in that a value of one indicates the assignment of flight *f* to slot *s* at airspace resource *i*. For this set however, the additional dimension *q* indicates the capacity scenario for which this conditional plan is developed. The existence conditions, shown in (5.2), are similar to the previous set of decision variables, in that they require the revised slot time to be later than or equal to the flight's scheduled arrival time.

$$y_{fs}^{qi} \quad binary \qquad \forall q \in Q, f \in F, i \in V_f, s \in Q_f^{qi} \tag{5.2}$$

Again, a range is employed to limit the existence of decision variables. Building on the earlier definition of feasible slots for each flight Q_f^i, the stochastic equivalent range Q_f^{qi} is defined in (5.3), conditional on the slot times for outcome *q*.

$$Q_f^{qi} = \left\{ s \in S^i : \tau_s^{qi} \geq \alpha_f^i \right\} \tag{5.3}$$

4.3.2 ***Initial decisions***

The constraints defining initial decisions in this formulation are equivalent to those for the deterministic formulation presented earlier. They are repeated here for clarity. Two sets of assignment constraints are employed to ensure feasible allocations at each resource. The first set, shown in (5.4), enforces the condition that each flight, under the initial plan, be assigned to exactly one slot at each resource $i \in V_f$ it will utilize.

$$\sum_{s \in Q_f^i} x_{fs}^i = 1 \qquad \forall f \in F, i \in V_f \tag{5.4}$$

Constraint set (5.5) enforces the first stage capacity constraints that each slot receives at most one flight.

$$\sum_{\substack{f \in F: \\ i \in V_f}} x_{fs}^i \leq 1 \qquad \forall i \in I, s \in S^i \tag{5.5}$$

The two constraint sets shown above, in isolation, will allocate flights to slots at only a single resource. The constraint set that links together these multiple resources is shown in (5.6). It enforces that condition that if slot *s* is chosen at resource *i*, then some slot *t* whose time falls in the range R_{fs}^{ij} must be chosen at resource *j*.

$$x_{fs}^i - \sum_{k \in R_{fs}^{ij}} x_{kt}^j \leq 0 \qquad \forall f \in F, i \in V_f, j = N_f^i, s \in Q_f^i : \left| N_f^i \right| > 0 \tag{5.6}$$

4.3.3 ***Revised decisions***

The constraints defining the first stage decisions are rewritten for each potential capacity outcome $q \in Q$ using the appropriate $\left\{ y_{fs}^{qi} \right\}$ variables in (5.7), (5.8), and (5.9).

$$\sum_{\substack{s \in S^i : \\ \tau_s^{qi} \geq \alpha_f^i}} y_{fs}^{qi} = 1 \qquad \forall q \in Q, f \in F, i \in V_f \qquad (5.7)$$

$$\sum_{\substack{f \in F : \\ i \in V_f}} y_{fs}^{qi} \leq 1 \qquad \forall q \in Q, i \in I, s \in S^i \qquad (5.8)$$

$$y_{fs}^{qi} - \sum_{k \in R_{fs}^{qij}} y_{fk}^{qj} \leq 0 \qquad \begin{array}{l} \forall q \in Q, f \in F, i \in V_f, j = N_f^i, s \in Q_f^{qi} : \\ \left| N_f^i \right| > 0 \end{array} \qquad (5.9)$$

The range defining the feasible arrival times at a subsequent resource is also rewritten to accommodate the additional dimension of indices used for each scenario outcome, shown in (5.10).

$$R_{fs}^{qij} = \left\{ k \in S^j : \max\left(\alpha_f^j, \tau_s^{qi} + \alpha_f^j - \alpha_f^i - \pi_L \right) \leq \tau_k^{qj} \leq \tau_s^{qi} + \alpha_f^j - \alpha_f^i + \pi_U \right\} \qquad (5.10)$$

4.3.4 ***Consistency***

The constraints defining both the initial decisions, as well as the revised decisions under the stochastic outcomes are, in isolation, equivalent to the deterministic problem shown in §2. Of course, for a model optimizing allocations simultaneously over multiple uncertain capacity outcomes to be useful, the various outcomes must be linked. The formulation presented here represents only a two stage decision process, hence only a single change in capacity is permissible. Two constraint sets are required to ensure consistency at the time of that change. The first fixes the values of first and second stage variables preceding the change to be equal, while the second ensures that the two allocations are consistent

The first of these consistency constraints is shown in (5.11). Assuming that it is possible for the flight to arrive before the revision time t^q, this constraint set fixes the values of the first and second stage decision variables to be equivalent. In principle, this constraint could be eliminated by defining the second stage variables to exist only after the revision time; however this would complicate the second consistency constraint, as well as the objective function.

$$x_{fs}^{i} = y_{fk}^{qi} \qquad \forall q \in Q, f \in F, i \in V_f, s \in G_f^{qi} \tag{5.11}$$

The range of slot times over which constraint (5.11) is defined is defined as G_f^{qi}, as shown in (5.12). It begins at the flight's arrival time and ends at the scenario realization, at which time consistency is no longer valid and a flight may be reassigned accordingly.

$$G_f^{qi} = \left\{ s \in S^i : \alpha_f^i \leq \tau_s^i \leq t^q \right\} \tag{5.12}$$

The second consistency constraint is shown in (5.13). It is logically more complex and requires greater explanation.

$$x_{fs}^{i} - \sum_{k \in T_{fs}^{qi}} y_{fk}^{qi} \leq 0 \qquad \forall q \in Q, f \in F, i \in V_f, s \in Q_f^i \tag{5.13}$$

Notionally, this constraint requires that all flights receive feasible revised slot allocations, in particular, that those flights that have not yet arrived by the revision time receive compatible assignments under the new allocation, with respect to their original allocations. For flights en route, this means that they not have to speed up or hold excessively. For flights still on the ground, this means that they not be assigned slot arrival times any sooner than the required travel time from origin to each resource.

The structure of this constraint is quite similar to the linking constraints employed to guarantee feasible slot allocations between subsequent airspace resources: the difference of a possible initial allocation and the sum of several possible secondary allocations must be nonpositive. In this case however, the critical difference lies in the range of feasible slots in the secondary allocation considered. This constraint is evaluated for each combination of flight and feasible first stage slot allocation for each feasible second stage allocation.

The range of feasible second stage reassignments T_{fs}^{qi}, conditional on a flight's first stage assignment, is precomputed. It must encompass feasible reassignments both for flights still on the ground, and for those already in the air. The consideration of these flight states, however, is not dynamic: each possible outcome is constrained separately by (5.13).

To illustrate qualitatively the ranges of slots to which a flight may be reassigned, consider the example shown in Figure 4-6. Eligible slots for the initial assignment are shown in the left column, while slots under the revised, second stage, assignment are shown in the right. In this case, under the revised scenario, the interarrival time has been decreased for some period under the revised plan. A flight is determined to be on the ground at the revision time t^q if its first stage allocation is at a time greater than the sum of the revision time and the flights required travel time. In that case, a flight may be reassigned as shown in Figure 4-6. In this example, if a flight was initially assigned to arrive to Slot 6, and is still on the ground because the corresponding departure time has not yet been reached, then in the new allocation, it may be assigned to any slot later than the sum of the current time, t^q, and the required travel time.

However, if a flight is initially assigned to arrive at a slot such that it must have already departed, then the range of slots to which it may be reassigned is likely smaller, as shown in Figure 4-7, because this reassignment-induced change must be absorbed while the flight is in the air. Of course, if interarrival times were to increase under the revision, it may be necessary to assign significant airborne delay to this flight.

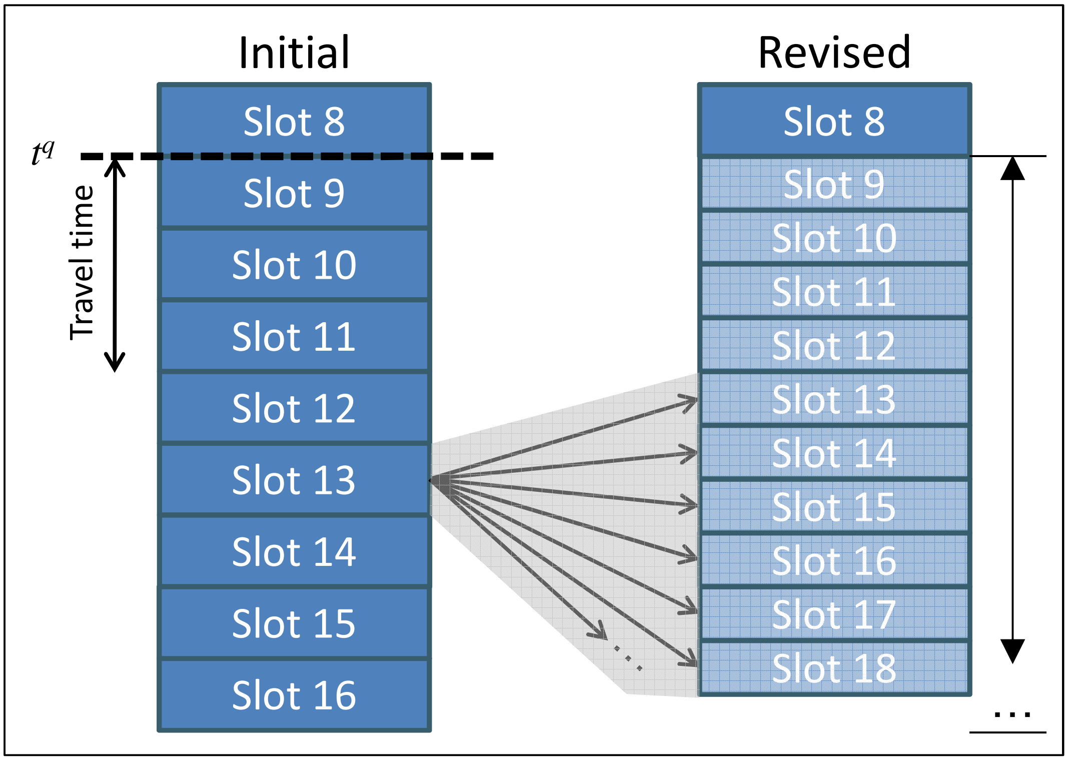


Figure 4-6 – Feasible reassignment range for flights on ground

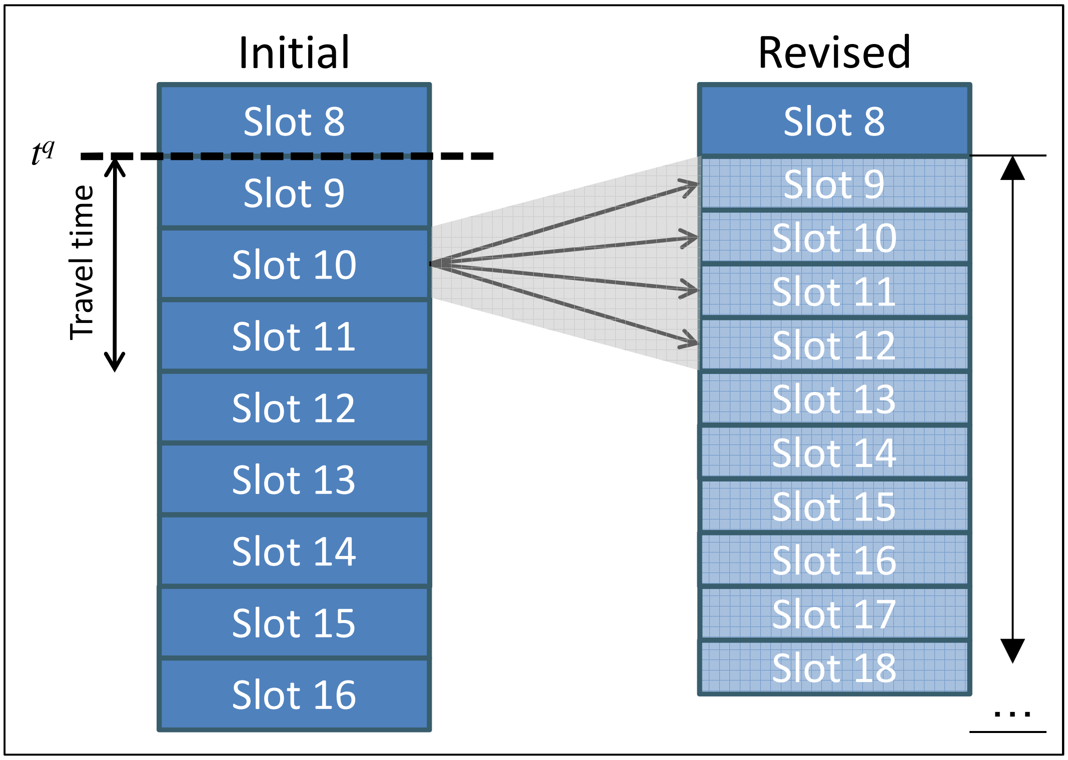


Figure 4-7 – Feasible reassignment range for flights en route

The range encompassing these two example conditions is shown in (5.14), but the reasoning and necessity underlying each term will be presented in the subsequent discussion, as a relatively high degree of complexity is incorporated. The difficulty in formulating this feasible slot time range lies in the fact that the flights being reassigned may, or may not, have already departed, depending on their initial slot assignment. The range of feasible slots for those flights still on the ground is much larger than for those in the air because en route flights carry a finite amount of fuel and thus cannot hold indefinitely. The upper and lower bounds in the range T_{fs}^{qi} are designed to accommodate this duality. Importantly, this process represents computations and procedures undertaken to generate inputs to the optimization formulation described.

$$T_{fs}^{qi} = \left\{ k \in S^i : \begin{array}{c} \max\left[\alpha_f^i, \min\left(\tau_s^i - \kappa_L, t^q + \alpha_f^i - \delta_f \right) \right] \le \tau_k^{qi} \le \\ \max\left[\tau_s^i + \kappa_U, M\left(\tau_s^i - \left(t^q + \alpha_f^i - \delta_f \right) \right) \right] \end{array} \right\} \tag{5.14}$$

4.3.4.1 Lower consistency bound

The lower consistency bound is developed to enforce the condition that, under a revised plan from this model, a flight may only be reassigned a certain amount earlier than it was under the initial plan. This allowable deviation depends on several factors, including whether or not the flight would already have taken off, had it been initially assigned to the slot being considered.

A flight *f* is deemed to have already departed in this model if the "current" time t^q is later than the difference between the considered slot *s* and the travel time required ($\alpha_f^i - \delta_f$) for flight *f*. The "current" time in this situation is the time at which the reallocation is made: the scenario realization time t^q. However, if the sum of the current time t^q and the required travel time is less than the slot *s* under consideration, then the flight is deemed to not yet have departed. To determine the lower bound of the feasible reassignment range, the minimum of these two quantities must be considered, as shown in (5.15). The parameter κ_L is included because, by assumption, an en route flight may be reassigned up to κ_L units earlier than originally planned, representing a speed increase or other actions to expedite the arrival of the flight. This parameter is analogous to the π_L used in the linking constraints in that it controls the maximum permissible increase in speed, but the two are not necessarily equal.

$$\min\left(\tau_s^i - \kappa_L, t^q + \alpha_f - \delta_f\right) \tag{5.15}$$

The expression (5.15) is nearly sufficient for the lower bound. However, by assumption, a flight may not be assigned an arrival time before its published scheduled time of arrival. Thus, the lower limit of this range is the expression in (5.15), or the flights scheduled time, whichever is greater, as shown in (5.16).

$$\max\left[\alpha_f, \min\left(\tau_s^i - \kappa_L, t^q + \alpha_f - \delta_f\right)\right] \tag{5.16}$$

4.3.4.2 Upper consistency bound

The upper consistency bound is constructed similarly. In this case, flights already en route may be assigned near to their originally assigned arrival time, or possibly much later if capacity conditions degrade significantly. Flights still on the ground, however, may be assigned as late as the end of the capacity rationing program.

To begin, a flight *f* is still on the ground if the condition shown in (5.17) is true. The condition defined here is such that the difference between now (t^q) and the slot being considered (τ_s^i) must be greater than the required travel time ($\alpha_f^i - \delta_f$).

$$\tau_s^i - \left(t^q + \alpha_f^i - \delta_f\right) > 0 \tag{5.17}$$

If this condition is true, then the flight is still on the ground, and the upper bound on the new slot assignment is the end of the assignment program. To allow for this, a large value M is multiplied with the value of this difference to form a large value for the bound. Note that this M value simply represents some very large number and is only used here in preprocessing to generate the feasible slot ranges. Its presence is not appreciated directly in the formulation, thus avoiding the numerical problems that often accompany using M in the traditional optimization sense.

However, if the difference shown in (5.17) is nonpositive, then the flight must have already departed. In that case, the upper bound on the new slot time must be the sum of the old slot time and some parameter κ_U to represent the maximum amount of slowing permissible for the flight. Given that the flight is already en route, the product of the large number M and the difference in (5.17) will be negative, and so using a maximum operator will select the correct value for these en route flights, as shown in (5.18).

$$\max\left[\tau_s^i + \kappa_U, M\left(\tau_s^i - \left(t^q + \alpha_f^i - \delta_f\right)\right)\right] \tag{5.18}$$

Further, it may be useful to make the reassignment window parameters κ_L and κ_U functions of the time remaining until the flight should arrive under the instance of the consistency constraint under consideration. This variation follows the idea that a flight located quite far from a rationing initiative has more time to increase or decrease speed, whereas one about to arrive at an initiative has very little flexibility about the time at which it is to arrive there.

A simple means by which this condition might be included is to specify the κ values as a monotonically decreasing function of the difference between the original and new slots, $\tau_s^i - t^q$. The functions $\kappa_L(t)$ and $\kappa_U(t)$ would be defined over the domain $\left(0, \alpha_f - \delta_f\right]$, and would take on zero values as $t = 0$ and much larger values at the upper end of the function's domain. This upper range could be as large as 45 minutes, as all commercial flights participating in such ATFM actions carry at least that much fuel in reserve.

4.3.5 *Objective function*

Generically, the objective of this formulation is to minimize the sum of assigned delays. There are two specific issues to be addressed in developing this objective function, however: how to incorporate the costs of each scenario outcome, and again at which resources to sum delays.

There are several potential methods by which the costs of the various recourse outcomes may be included. Based on the assumption presented earlier that the initial plan is always included as a second stage outcome with non-zero probability; all costs may be represented in the second stage. As a result, an expected value of the total cost may be calculated using these costs and the associated scenario probabilities. This is notationally simpler than the alternate convention of expressing first stage costs and second stage marginal costs.

The second issue in developing the objective function again concerns which delays to include in the sum. As was discussed in detail in §2, reasonably arguments can be made for considering the sum of delays at all resources, or only those delays at arrival airports. Both will be examined as to their effects on this problem under capacity uncertainty.

The total delay objective function for this problem is shown in (5.19). It is expressed as the expected value of the sum of all second stage allocations. It represents the ground delays assigned to each flight under each scenario outcome, with the length of the delay represented by the difference between the assigned slot and the scheduled time. A superlinear function of delay length is again employed to encourage more equitable distribution of delays. To consider only arrival delays at destination airports, the condition $\left|N_f^i\right| = 0$ must be added to the summation in V_f.

$$\min z = \sum_{q \in Q} p^q \sum_{f \in F} \sum_{i \in V_f} \sum_{s \in Q_f^i} \left(\tau_s^{qi} - \alpha_f^i \right)^{1+\varepsilon} y_{fs}^{qi} \tag{5.19}$$

4.3.6 *Formulation size*

As with the previous optimization models presented, the formulation size is considered here. Table 4-4 includes the worst case numbers of constraints for each set, while Table 4-5 lists the worst case numbers of variables for each set of decision variables.

Constraint	Worst case count
(5.4)	$\sum_{f \in F} \lvert V_f \rvert$
(5.5)	$\sum_{i \in I} \lvert S^i \rvert$
(5.6)	$\sum_{i \in I} \lvert S^i \rvert \sum_{f \in F} \lvert V_f \rvert$
(5.7)	$\lvert Q \rvert \sum_{f \in F} \lvert V_f \rvert$
(5.8)	$\lvert Q \rvert \sum_{i \in I} \lvert S^i \rvert$
(5.9)	$\lvert Q \rvert \sum_{i \in I} \lvert S^i \rvert \sum_{f \in F} \lvert V_f \rvert$
(5.11)	$\lvert Q \rvert \sum_{i \in I} \lvert S^i \rvert \sum_{f \in F} \lvert V_f \rvert$
(5.13)	$\lvert Q \rvert \sum_{i \in I} \lvert S^i \rvert \sum_{f \in F} \lvert V_f \rvert$

Table 4-4 – Worst case number of constraints for stochastic linked formulation

Variable	Worst case count
$\left\{ x_{fs}^i \right\}$	$\sum_{f \in F} \sum_{i \in V_f} \lvert S^i \rvert$
$\left\{ y_{fs}^{qi} \right\}$	$\lvert Q \rvert \sum_{f \in F} \sum_{i \in V_f} \lvert S^i \rvert$

Table 4-5 – Worst case number of variables for stochastic linked formulation

Although the results in the above tables appear extremely large, in realistic instances, they are considerably smaller. The various sets limiting the ranges over which constraints and variables are defined ensure this. However, as with most stochastic integer programs, formulation size remains a concern.

4.3.7 *Generalized model of capacity evolution*

The model presented above assumes that uncertainty will admit only improved capacity conditions, by examining the range of possible early end times for a disruption. To examine more general cases, several changes must be made to this formulation. These are presented separately because they may increase significantly the complexity of the model, and because they are not used in the case study examined in the next section.

To make this model compatible with more general models of capacity evolution, several changes are needed. First, the parameters of the consistency range defined previously must be specified so as to allow potentially large airborne delays. These may be necessary in the even that capacity conditions deteriorate significantly. However, when potentially lengthy airborne delays are admitted, the overriding assumption of the small time deviations permitted by the linking and consistency constraints becomes tenuous. To that end, a mechanism must be included for tracking the cost of these delays.

To track the cost of airborne reassignment delays, a third set of decision variables is introduced. This set $\left\{z_{fsk}^{qi}\right\}$ is employed to track slot reassignment of flights already en route to their destination. These are flights that have already departed when a new scenario is realized that requires the flights to be reassigned with potentially significant airborne delays. The indices show that flight *f* must be reassigned from slot *s* to slot *k* under capacity outcome *q* at airspace resource *i*, as shown in (5.20). The variables have a value of unity when this condition is realized and zero otherwise. They are defined over some precomputed range U_{fs}^{qi} that identifies flight-slot-scenario combinations that might yield airborne holding.

$$z_{fsk}^{qi} \;\; binary \qquad \forall q \in Q, f \in F, i \in V_f, s \in Q_f^i, k \in U_{fs}^{qi} \quad (5.20)$$

These new decision variables represent the confluence of two conditions: that a flight already en route was initially assigned to slot *s* at resource *i and* that a flight was, under outcome *q*, reassigned to slot *k* at resource *i*. This can be formulated as a logical AND constraint, and could be incorporated by examining the product of the two decision variables corresponding to the above assignments. However, to maintain linearity of the formulation, the construct shown in (5.21) was employed. These three constraint sets, employed together, set the value of z_{fsk}^{q} equal to the logical AND value of x_{fs}^{i} and y_{fk}^{qi}, constrained to lie within some range U_{fs}^{qi}.

$$\begin{aligned} z_{fsk}^{qi} &\le x_{fs}^{i} \\ z_{fsk}^{qi} &\le y_{fk}^{qi} \\ z_{fsk}^{qi} &\ge x_{fs}^{i} + y_{fk}^{qi} - 1 \end{aligned} \qquad \begin{aligned} &\forall f \in F, i \in V_f, s \in S^i, k \in S^i, q \in Q: \\ &\tau_s^i \ge \alpha_f^i, \tau_k^{qi} \in U_{fs}^{qi} \end{aligned} \qquad (5.21)$$

Once the appropriate decision variables have been defined, a cost must be assigned to them. To this end, a second term is added to the expectation shown in (5.19). This function, shown in (5.22), represents the cost of airborne delays introduced as a result of flights receiving slot reassignments at the realization of a new capacity scenario. The parameter ϕ represents the cost ratio of airborne to ground delays, because reassignment delays are realized by flights already en route. While the reassignment delay may be either positive or negative, depending on whether the flight was given and earlier or later slot, only the magnitude is considered. Again, a superlinear function of delay length is employed. The cost of early and late “delays” are treated here as being equivalent, although in practice different values may be assigned to each of these. The range L_f^{qi} is defined such that only flights that could already be en route for slot *s* under scenario *q* are included in the summation.

$$\Omega_2 = \sum_{s \in L_f^{qi}} \sum_{k \in Q_f^i} \phi \left| \tau_s^{0i} - \tau_k^{qi} \right|^{1+\varepsilon} z_{fsk}^{qi} \tag{5.22}$$

4.4 Case study

To test the effectiveness of the model proposed in this chapter, a realistic case study is examined here. The intent of this case study is to consider the output of this stochastic model to identify trends and patterns in the proscribed allocations. The same physical configuration and schedule data are used as in the earlier case studies, but the capacity data clearly must be changed to reflect the multiple possible outcomes.

4.4.1 *Stochasticity in capacity data*

Stochasticity in the capacity data is introduced by varying only a single parameter of the simple model of capacity used in this chapter. For this case study, the effect of uncertainty regarding disruption end time at each resource is examined. The disruption is initially planned to last 120 minutes at resource A and 150 minutes at resources B and C. Two additional scenarios are included with the disruption ending either 30 or 60 minutes early. Only the airspace region (A) and one airport (B) are assumed to have variable disruption end times. Airport C is assigned a fixed duration of 150 minutes, to help allow the examination of the value of the stochastic scenarios, as well as to prevent symmetry in the case study. Each of the early end times is assigned probability 0.3, while the initially planned duration is assigned 0.4. The scenarios are realized jointly, that is, both resources A and B end early by the same amount, or neither does. These resource capacities are illustrated in Figure 4-8.

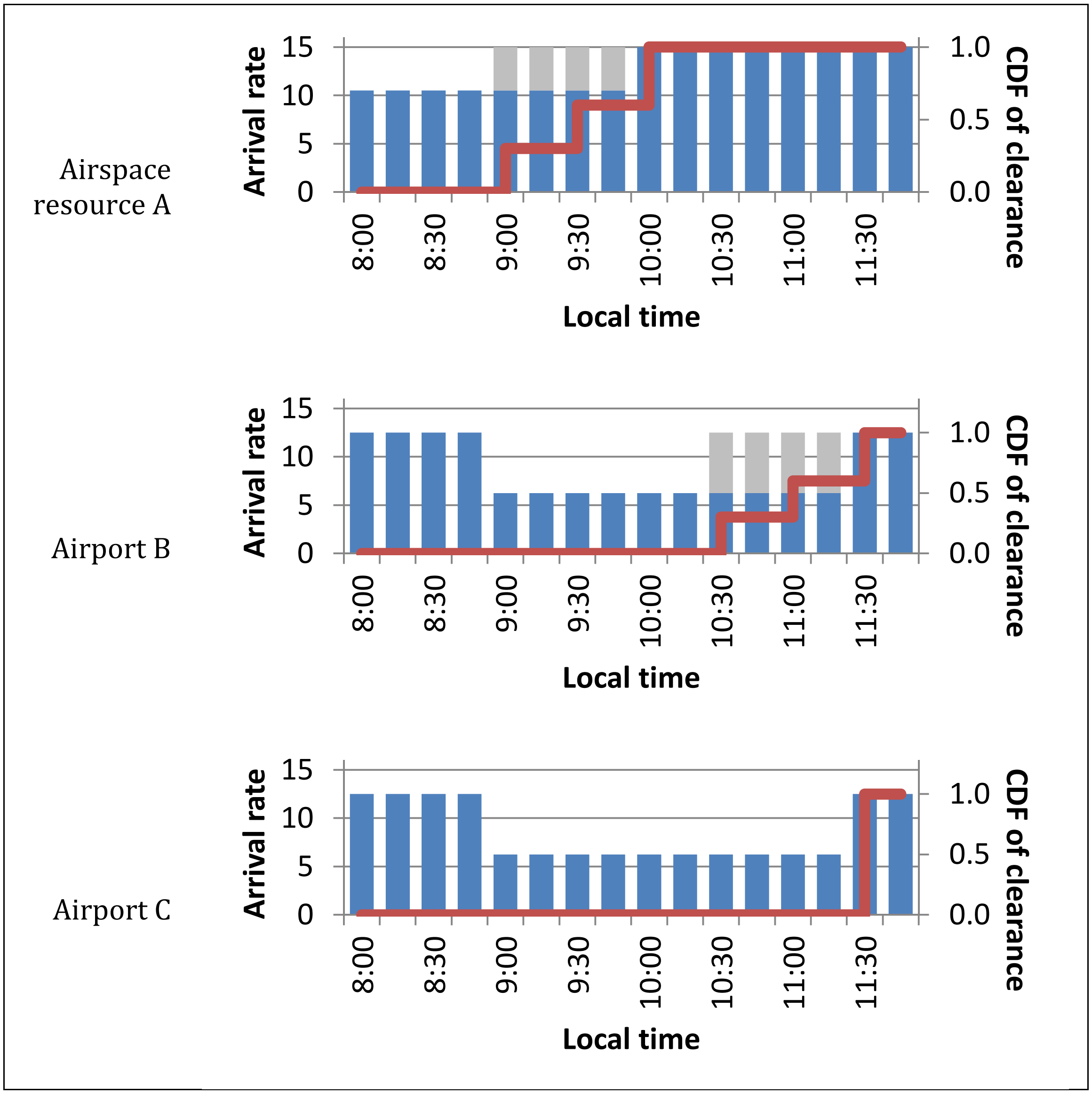


Figure 4-8 – Capacity scenarios for each resource

Two variations on the case study are used to examine the effect of scenario realization times. In the first, the capacity increase is not anticipated – the end time of the disruption, or the onset increased capacity, is unexpected. In the other case, 30 minutes after the beginning of the disruption, the remaining evolution of the capacity becomes clear – this represents improved predictive ability or weather forecasting technology. Unless otherwise noted, the results presented represent the case in which this lookahead ability is not present.

Several categories of results are presented to illustrate the power of this model. They are intended to evaluate both the properties of this model, as well as the value of stochastic information in the coordinated ATFM process. Each set of results will be discussed in a separate subsection.

4.4.2 *Computational testbed*

The computational experiments in this case study were performed using powerful computer hardware and software. The system used has four dual-core Intel Xeon X5355 processors and 12GB of memory. It runs software in a 64-bit environment under Windows Server 2003 Enterprise edition.

The optimization tests were conducted using Fair Isaac's Xpress 2008b 64-bit software. Models were coded using the Mosel language and executed through Xpress' graphical interface, Xpress-IVE. The rule-based approaches were coded in MATLAB R2008a running on the same hardware.

4.4.3 *Summary results*

The first set of results in this case study considers the aggregate performance of this formulation using both the total and final delay objective functions according to both the total and final delay metrics, as shown in Table 4-6.

Metric	**Objective**	**Assigned delays (minutes)**				
		Expected	*Initial plan*	*Scen. 1*	*Scen. 2*	*Scen. 3*
Total delay (minutes)	Total	14894	16373	12952	14865	16373
	Final	**14027**	16182	11165	14380	15908
Final delay (minutes)	Total	11246	12223	9946	11244	12223
	Final	**8619**	9813	7272	8738	9540

Table 4-6 – Comparison of assigned delays

Several trends are apparent in this data. First, the allocation developed by the final delay objective dominates that from the total delay objective, as both metric are minimized by the final model in expectation as well as for each scenario.

However, this trend is explained by two phenomena. It is important to note that the two solutions represented here, evaluated according to both the total and final delay metrics, do not represent optimal solutions, although the optimality gap in each case was fairly small. In addition, the data shown in Table 4-6 describe the properties of the allocations themselves – they do not represent the functional values being minimized.

However, despite this initial quirk, the appropriate pattern in the delays assigned under each scenario is observed for each case. The several scenarios represent early end times, and as such result in some delay savings.

This aggregation represents only one possible examination of the data. In the following sections, different aggregations will be examined to identify trends and draw conclusions about the efficacy of these various models.

4.4.4 *Comparison to deterministic results*

After confirming that the stochastic models proposed here generally perform as expected, the next issue of interest lies in comparing their performance with that of the models considering deterministic capacity. In Table 4-7, the expected values of delays assigned by the stochastic models are compared to the values of delays assigned by the deterministic models. Again, both objective functions and both metrics are considered.

Metric	**Objective**	**Capacity model**	
		Deterministic	*Stochastic*
Total delay (minutes)	Total	**14169**	14894
	Final	14439	**14027**
Final delay (minutes)	Total	**10723**	11246
	Final	8929	**8619**

Table 4-7 – Comparison of deterministic and stochastic model results

Again, the stochastic model using the total delay objective function performs relatively poorly, for the reasons outlined above. For both metrics, it produces the poorest results. Only expected results are compared here, however if these numbers are compared to those for each outcome in Table 4-6, then it becomes clear that the total delay objective with stochastic capacity may produce superior allocations, given the recourse actions used. The stochastic model using the final delay objective takes full advantage of the recourse made available and produces allocations with superior properties in expectation to all those others.

4.4.5 *Comparison of treatment of various flows*

The next set of analyses examine whether the two objective functions continue to exhibit the same biases under stochastic capacity assumptions as they did under deterministic. The total delay model prefers to maintain schedule order, while the final delay model prefers to prioritize flights using fewer resources. Table 4-8 examines the nature of the solution with respect to the number of resources used. This provides some measure of the equity between flights.

Number of resources	**Number of flights**	**Obj.**	**Arrival delays (minutes/flight)**				
			Exp.	*Initial plan*	*Scen. 1*	*Scen. 2*	*Scen. 3*
1	177	Total	41.8	45.3	37.1	41.7	45.3
		Final	18.3	19.2	19.3	17.4	18.2
2	95	Total	40.6	44.2	35.6	40.6	44.2
		Final	56.6	67.6	40.5	59.6	66.5

Table 4-8 – Comparison of delays according to number of resources used

This table makes it clear that the same property extends to stochastic capacity assumptions. The delays assigned to single resource flights by the final delay model are markedly lower than those assigned for multi-resource flights. In contrast, the delays assigned by the total delay model to both single and multi-resource flights are very similar.

The comparison of delays assigned to flights using either one or two resources is only a partial view of the bias and equity properties of these models. A more specific examination of equity and stochasticity is depicted in Table 4-9, with the expected minutes of delay per flight shown for each flow. This division of flights follows the flows defined in Figure 2-17.

Flow	Number of flights	Objective	
		Total	*Final*
1	58	40.2	8.0
2	48	36.7	54.3
3	59	40.9	20.4
4	47	44.5	59.1
5	60	44.2	26.1

Table 4-9 – Results according to flow and capacity scenario

The same trends identified above, as well as in §2, continue in these results. Flows 2 and 4 are assigned uniformly larger delays by the final delay model, while flights 1, 3, and 5 receive smaller delays. Flows 4 and 5 do receive larger delays in expectation. This trend will be explored further in the next section.

4.4.6 *Comparison of delays by destination*

When specifying the stochastic capacity scenarios for this case study, airport C was chosen to lack variability. This represents the condition in which either no information about alternate outcomes is available, or in which conditions are known with certainty, and so no random outcomes need be considered. In the previous analysis, it seemed that flows 4 and 5, destined for airport C, received larger delays. In this section, this trend is examined in greater detail.

One measure of the value of stochastic capacity information is provided in Table 4-10. Resources A and B have stochastic capacity descriptions, while resource C does not – no early end times are considered there. It is clear that flights destined for airport C receive generally larger delays than those destined for airport B. The delays for flights destined to resources A and B may be reduced by the possible early end times in scenarios 1 and 2. However, because there is no possibility of early an early ending to the disruption at airport C, delays assigned under both the initial plan and realized under each scenario are equivalent.

Destination	**Number of flights**	**Obj.**	**Arrival delays (minutes/flight)**				
			Exp.	*Initial plan*	*Scen. 1*	*Scen. 2*	*Scen. 3*
A	58	Total	40.2	44.2	34.9	40.1	44.2
		Final	8.0	10.8	2.0	10.4	10.8
B	107	Total	39.0	46.0	29.7	39.1	46.0
		Final	35.6	44.1	26.4	35.6	42.6
C	107	Total	44.3	44.3	44.3	44.3	44.3
		Final	40.6	41.8	40.5	40.5	40.7

Table 4-10 – Comparison of delays according to destination

4.4.7 *Effect of flight length on model results*

Another bias that may be introduced when considering stochastic capacity outcomes is that against flights of different durations. In Table 4-11 and Table 4-12, the effect of flight distance on model results is examined from the total and final delay objective functions, respectively.

For the total delay model, it is important to note here that shorter flights experience greater decreases in assigned delay with improved capacity conditions (earlier scenarios), both in relative and absolute terms, compared to longer flights. This comes as a result of the Ration By Distance principle explored in (Ball, Hoffman and Mukherjee 2009). It appears that this property extends to the multiple resource case when using the total delay objective; however quantifying this property for this more general model poses a considerably greater challenge.

The results from the final delay model are somewhat different. The same trend of increased savings with earlier end time scenarios is again observed. However, in this case, shorter flights tend to receive shorter delays. Likely this arises as a result of the correlation between the length of a flight and the number of resources that it can feasibly visit. Shorter flights are naturally more likely to visit fewer resources; ergo the trend observed for decreased expected delays for shorter flights is another reflection of the model's bias against multi-resource flights.

Destination	Number of flights	Arrival delays (minutes/flight)				
		Expected	*Initial plan*	*Scen. 1*	*Scen. 2*	*Scen. 3*
30	61	46.3	50.8	40.2	46.4	50.8
45	89	44.4	48.5	39.1	44.3	48.5
60	62	38.1	41.7	33.4	38.0	41.7
75	20	35.3	37.0	32.9	35.5	37.0
90	19	36.8	38.5	34.5	36.8	38.5
105	7	24.5	25.1	23.0	25.1	25.1
120+	14	37.9	41.3	33.6	37.6	41.3

Table 4-11 – Delays by distance category for total delay model

Destination	Number of flights	Arrival delays (minutes/flight)				
		Expected	*Initial plan*	*Scen. 1*	*Scen. 2*	*Scen. 3*
30	61	19.2	22.4	20.6	17.4	19.6
45	89	32.2	34.9	30.1	31.4	34.3
60	62	35.1	42.1	25.4	36.3	41.6
75	20	34.5	40.4	24.8	37.1	39.8
90	19	37.1	42.4	26.8	40.8	42.2
105	7	40.5	44.5	30.5	45.2	44.4
120+	14	51.8	57.5	38.7	57.5	57.3

Table 4-12 – Delays by distance category for final delay model

Another method of visualizing these results is shown in Figure 4-9. In this figure, the delay savings under each early end time are depicted, according to flight distance category. The results shown in this manner clearly indicate that shorter flights realize greater savings upon early end times, while longer flights realize smaller savings, in keeping with the RBD principle. The 120+ category represents an anomaly, likely due to the small number of flights in this grouping.

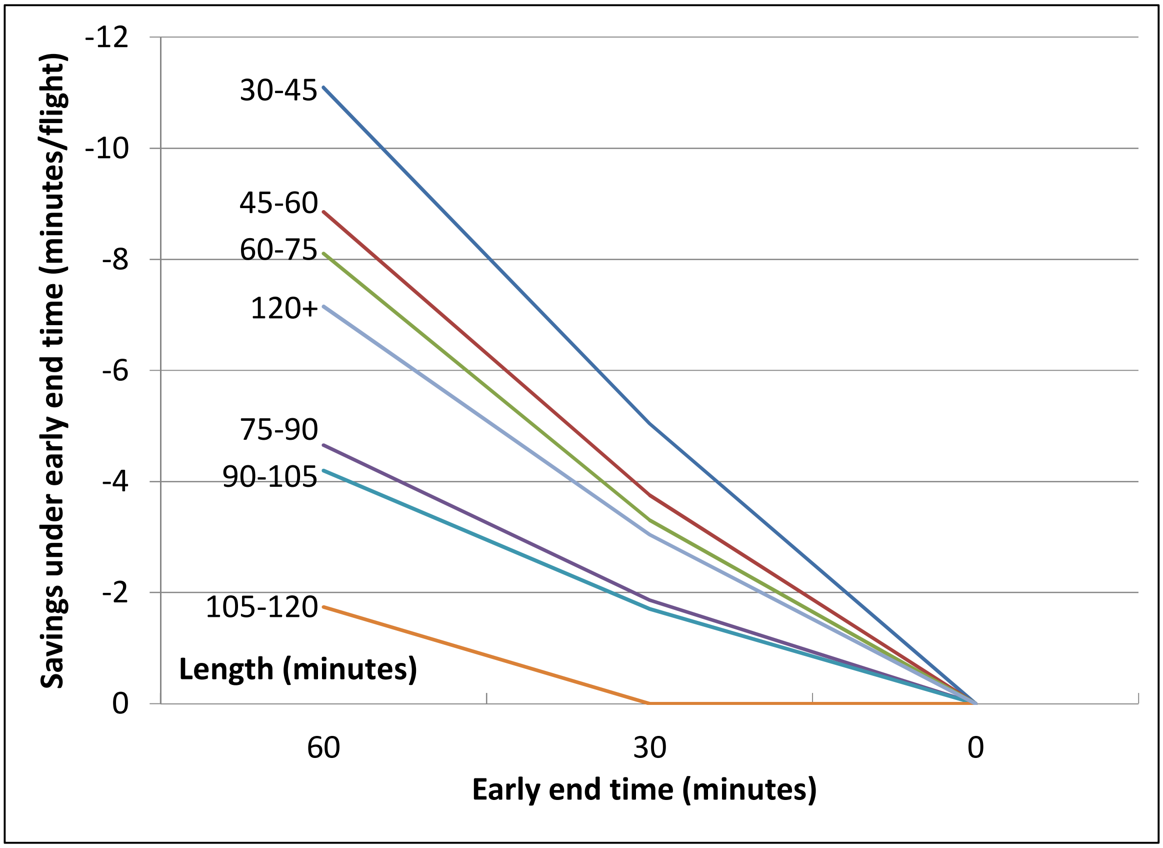


Figure 4-9 – Delay savings under early end times by distance

4.4.8 *Value of lead time in making decisions under uncertainty*

In Table 4-13, the value of forecast lead time is examined using a comparison of two cases for each objective. The only difference between these two cases is the time at which the capacity variation becomes known, t^q. In the case without lookahead, the change to a scenario is known only upon its occurrence. With lookahead, the true scenario is known at after 30 minutes of disrupted conditions.

The obvious trend in these results is that having lookahead ability earlier, or gaining knowledge about the capacity scenario to be realized, has value in reducing delays. However, it is also clear that this benefit is greater for allocations made under the total delay objective function that for those made under the final delay objective. This reflects the differing priorities of the models in making allocations.

Case	Obj.	**Arrival delays (minutes/flight)**				
		Expected	*Initial plan*	*Scen. 1*	*Scen. 2*	*Scen. 3*
Without lookahead	Total	41.3	44.9	36.6	41.3	44.9
	Final	31.7	36.1	26.7	32.1	35.1
With lookahead	Total	38.1	41.6	33.3	38.1	41.6
	Final	31.3	36.4	26.6	31.5	34.6

Table 4-13 – Effect of varying capacity scenario realization time

4.4.9 *Computational performance*

A summary of the computational performance for each of the four cases described here is shown in Table 4-14. Each run was terminated upon finding the first integer feasible solution, however in each case this solution was of reasonable quality. No special routines were employed in solving these instances – only branch and bound was used.

Objective	Lookahead	Solution time (seconds)	Gap from best bound
Total	Without lookahead	4226	2.8%
Total	With lookahead	1074	0.4%
Final	Without lookahead	33537	2.3%
Final	With lookahead	15433	3.4%

Table 4-14 – Computation performance

It is obvious from these results that the formulations encompassing stochastic capacity are more difficult to solve than the deterministic models shown earlier, even for the modest sized problem examined in this case study. This computational performance reflects a grave challenge to the utility of these models. The implications and strategies for mitigating this will be explored in the next section.

4.5 Practical considerations

There are many reasonable considerations about practical applications of the stochastic model for coordinated capacity allocation presented in this chapter. In this section, some of these will be specifically highlighted and discussed. As before, many of the previously-mentioned considerations relating to employing optimization in a practical setting continue to apply.

Complex integer programs reflecting stochastic systems face challenges both in development and wider acceptance. Primarily, they are limited by structural complexity, resistance from system users, and poor availability of useful stochastic description of capacity evolution. The first limitation may require considerable mathematical modeling efforts to overcome, and doing so has a distinct value in and of itself (Glover and Ball 2010), (Rios and Ross 2008), (Rios and Lohn 2009). As a practical concern, the size of realistic problems, reflected both in the number of flights, as well as the number of joint scenarios that may be examined is certainly constrained by the strength of the formulation.

One important concern in using a decision-making model with recourse is the application of the conditional plans developed for each outcome. The severity of the impact of this concern depends primarily on the hypothetical policy used to implement these plans. A distributed and collaborative system, as is operated today might only make use of the initial plan for allocations, and would allow airlines to make whatever plans for their individual flights upon the realization of any subsequent capacity changes. Conversely, in a system that did not foster such collaboration, decisions from this model could be used to dictate all operations. This would represent a severe and likely unrealistic change from today's operational paradigm.

Fortunately, there is precedent for an intermediate solution. Some research and development efforts for the Next Generation Air Traffic System (NextGen) have been devoted to implementing conditional plans input by airlines (Metron Aviation 2009). Under this paradigm, airlines could submit plans corresponding to each of the discrete capacity scenarios. The system operator could use a stochastic model as presented here to make the initial allocations, and to recommend optimal recourse plans. Airlines would not be bound by these recommendations of course, and would be able to submit their own conditional plans.

The third limitation identified here is the limited availability of information about uncertainty in capacity forecasts. The utility of the stochastic model presented in this chapter hinges on the availability of stochastic descriptions of airspace and airport capacities. The quality and availability of such data should be greater for airports than for arbitrary airspace resources. Airport capacity constraints are easier to characterize because the airport system is better bounded and the constraints that define the capacity itself (physical separation of aircraft) are more concrete. Airspace regions are more challenging for several reasons. First, it is more difficult to sense and quantify the weather conditions in them because of their distributed nature. Second, even if the weather conditions are well understood, the meaning of capacity is much more notional, given the large separations used in practice between aircraft and between disparate routes that may use the same region of disrupted airspace.

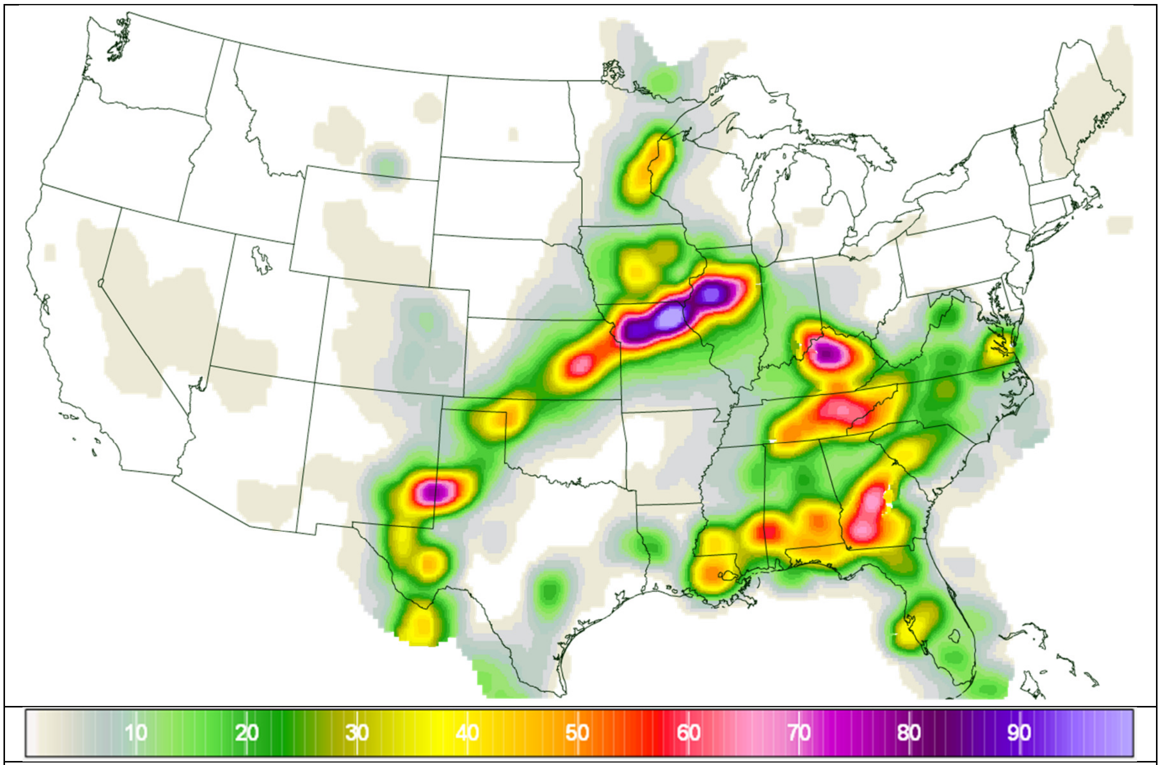
Some research has been conducted to this end, primarily on characterizing the stochastic nature of the airport capacities.. Robust statistical techniques have been used to develop scenarios based upon very short term forecasts (Buxi and Hansen 2010), for longer term trends (Liu, Hansen and Mukherjee 2008), or for very specific situations such as San Francisco's marine stratus layer (Cook and Wood 2009). Developing stochastic characterizations of capacity disruptions is an active and important area of research in ATFM.

The model presented here is particularly difficult, however, because it requires that scenarios for capacity outcomes be specified jointly for the several resources considered. Thus, in approximating some joint distribution of outcomes, there will be some loss in fidelity, or an increasingly large number of scenarios needed to accurately represent capacity outcomes.

However, despite these challenges for this model, advancements have been made in quantifying the weather uncertainty for arbitrary airspace resources. These do not necessarily incorporate the step of translation to meaningful capacity numbers, but represent an essential input to models such as that presented here. Models that quantify uncertainty surrounding convective weather are typically based primarily on numerical, statistical, or expert guidance methods. Numerical methods provide inputs for some statistical or expert-guided models.

Numerical weather forecasts utilize complex dynamical models of the atmosphere to produce forecasts of weather outcomes. The most relevant model is the Rapid Update Cycle (RUC). The underlying structure of the latest version of this model is described in (Benjamin, Dévényi, et al. 2004) and (Benjamin, Grell, et al. 2004). Multiple runs of this model are utilized to develop a probabilistic estimate for convective activity. This output is known as the RUC Convective Probability Forecast (RCPF), as described in (Weygandt and Benjamin 2004) and (S. S. Weygandt, et al. 2008).

One statistical weather model shown here is the Localized Aviation Model Output Statistics Program (LAMP). This model is based upon the Model Output Statistics (MOS) technique in which the results from numerical weather prediction models are processed with regression models. The numerical models do a good job predicting large-scale weather patterns and the regression models on their output are used to correct for variations in surface weather. The latest iteration of the LAMP model was proposed in (Ghirardelli 2005). The LAMP model produces many statistical forecasts of weather activity, including ceiling (Weiss and Ghirardelli 2005), winds (Wiedenfeld 2005), and thunderstorm probability (Charba and Feng 2005). An example of the thunderstorm probability data is shown in Figure 4-10.

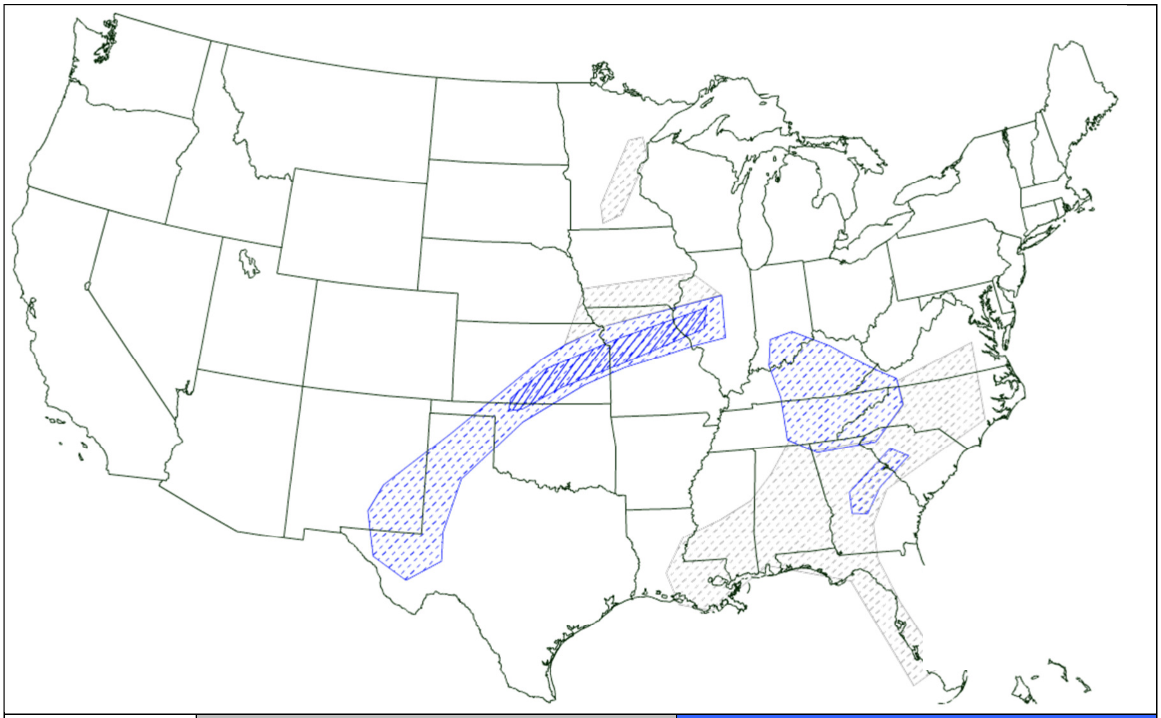


May 15, 2009 issued at 2100Z for +1-3 hours ahead
(from AvMet Applications Inc.)

Figure 4-10 – Sample LAMP thunderstorm probability

Finally, some probabilistic weather products reflect a combination of the above models with expert guidance. These include the Collaborative Convective Forecast Product (CCFP), which shows the expected occurrence of convection at two, four, and six hours ahead of the issuance time, and the Experimental Enhanced Thunderstorm Outlooks (EETO), which depicts contours representing regions of equal probability of convective activity.

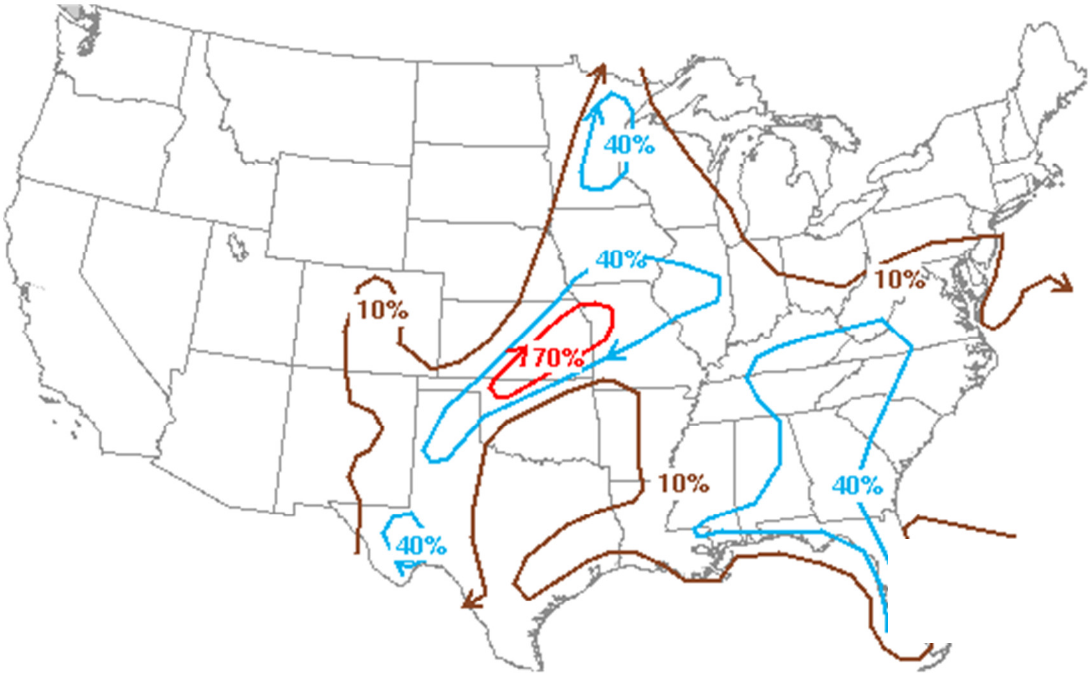
The CCFP is specifically designed to be used for strategic planning for en route operations in ATFM (Aviation Weather Center 2005). This differentiates it from some other products that are designed to be applied in the terminal area or for tactical decisions. Regions of airspace are included in the CCFP if they meet conditions about size, coverage density, cloud tops, and forecaster confidence score. Information about expected movement of the region may also be included. An example of several CCFP regions is shown in Figure 4-11.



Confidence:	25-49%		50-100%
Coverage:	Sparse: 25-49%	Medium: 50-74%	Solid: 75-100%
May 15, 2009 issued at 2100Z for +2 hours ahead (from AvMet Applications Inc.)			

Figure 4-11 – Sample CCFP regions

The EETO is also intended for strategic use, given the lengthy time horizon and limited spatial and temporal resolution (Storm Prediction Center 2009), limiting its utility for tactical operations. The contours are identified by expert forecasters using a variety of observations and numerical predictions to guide their assessments. An example of the output of this forecast is shown in Figure 4-12.



May 15, 2009 issued at 1635Z for 2000Z-2359Z
(from National Weather Service Storm Prediction Center)

Figure 4-12 – Sample EETO forecast

From this discussion, it should be clear that one of the primary concerns in using stochastic models such as that presented here for practical ATFM systems is the availability of data characterizing weather uncertainty. This is an active research area, and one in which advancement is essential to permit the development of more sophisticated decision support tools.

4.6 Conclusions

In this chapter, the problem of coordinating flight to slot assignments in multiple congested resources under uncertain capacity conditions is considered. First, the impact of stochastic variations in several capacity parameters is quantified. The results of this simulation showed that the model being evaluated is quite sensitive to some capacity parameters. Results indicate that model sensitivity, measured in terms of elasticity of assigned delay with respect to variations in capacity, decreases with model size, measured in number of resources considered.

However, in most cases, the assigned delay is elastic with respect to capacity variations, in that a unit change in the capacity parameters yields more than a 1% change in assigned delay.

That this ATFM model is so sensitive to variations in some capacity parameters suggested that more robust solutions may be derived through the use of a model that explicitly considers all possible modes of stochastic variation. To this end, a two stage formulation with recourse was introduced. Given a set of resources expected to be congested, for example several airports and an airspace region, and the set of flights expected to use those resources over some time horizon over several hours, a coordinated matching of flights to slots is developed.

Using a realistic, but artificially generated case study, this model was evaluated. The results of this analysis demonstrate this model functions as expected. Delays are reduced versus the deterministic analog, and are assigned, to some degree, equitably. The Ration By Distance principle is observed, as short flights are held back to provide a reserve pool of flights able to be dispatched and take advantage of newly available capacity.

5. Conclusions

This dissertation has examined the problem of coordinating capacity allocation between several resources in the airspace system, including multiple airports and airspace regions. Current practice, and many research efforts, treats each congested resource in the airspace system as an entirely independent capacity allocation problem. In reality, these resources are connected by flights that use multiple of them in sequence. Recognizing the conflict that may exist in solving these capacity allocation problems independently, several approaches were proposed in the three technical chapters of this dissertation.

The first set of approaches to develop coordinate capacity allocation plans were presented in §2. There, an integer optimization model was first described, and the properties of its resulting allocations analyzed. This formulation was unique in the problem scope that it approached, as well as in the representation of time-varying capacities using time slots, rather than aggregate time bins. The analysis of the model allocations for two different objective functions demonstrated that optimizing a superlinear function of the arrival delay of each flight yielded allocations that disadvantaged flights using multiple resources whenever it was possible to prioritize several flights using fewer resources. The essence of this bias is derived from the measurement of delay taking place only upon arrival. In contrast, minimizing the sum of all "delays" assigned at each resource, whether they were realized as arrival delays or not, guaranteed maintenance of the flights schedule order.

Recognizing these implications, and several of the general problems associated with employing optimization models in a practical distributed system such as this, rule-based heuristic solution techniques were next examined. Two different priority schemes were evaluated – one of these mimicked the analytic principles of the optimization model, and the other prioritized individual resources over others to mimic the operational systems employed today. Results suggested that each of these models was able to generate quality allocations in very little time.

Building further upon the results of the analysis in §2, several variations on this base optimization model were proposed in §3. In this chapter, several approaches were presented to explicitly control the equity properties of the resulting allocations. These efforts were undertaken to mitigate the phenomenon demonstrated earlier, wherein one class of users (multi-resource flights) may be disadvantaged at the expense of another (single resource flights). Both constraints on the maximum permissible deviation from the fair allocation, as well as cost-based approaches were considered.

The results of these equity models introduced several interesting trends. First, computational performance was improved in most cases, while still deriving desirable and efficient allocations. However, the purpose of these models was to improve the equity properties of the resulting allocations, and this was also successful. One important result from this analysis demonstrated the optimal solution when admitting flights-slot assignments only at or after the agreed earliest fair arrival time (that from the RBS allocation) did not vary significantly with the maximum deviation permitted. That is, when only such negative deviations were permissible, there was nearly no marginal benefit to increasing the slack the model was permitted. Thus, once this minimum feasible deviation has been identified, it seems that the optimal strategy, both to maximize efficiency (minimize average arrival delay) and to maximize equity (minimize maximum deviation) is to set the maximum deviation parameter equal to its minimum value.

The first two technical chapters provided a comprehensive overview of methods and issues in solving coordinated capacity allocation problems under deterministic capacity. In §4, the problem was reexamined with relaxed capacity assumptions. First, the expected impact of variations in capacity was quantified through an analysis that combined simulation and optimization techniques. This study demonstrated that the basic optimization model proposed for this problem is quite sensitive to variations in capacity, often yielding several percentage points increase in average delay for a single percentage point change in some capacity parameters.

Thus, recognizing that this problem itself is sensitive to uncertainty in capacity values, an optimization model was proposed that explicitly included this information. This two-stage model with recourse made use of one set of the same constraints as for the deterministic formulation for each capacity scenario, and added linking constraints to ensure that feasibility was maintained across the scenario realization boundary. The model was described in the context of modeling uncertainty surrounding the end time of a capacity disruption, but generalizations were proposed that would admit a greater range of possible capacity variations.

The case study for this stochastic model indicated that more robust allocations could be developed by explicitly considering uncertainty surrounding the end time of a capacity disruption. Because the model provides recourse actions, there is a mechanism to anticipate capacity increases and subsequently take advantage for them when they are realized. In addition, the model results demonstrated that the Ration By Distance principle for minimizing expected delays under uncertainty seems to extend to the multi-resource case examined in this dissertation.

There are several directions for interesting avenues of continued research that may be derived from this dissertation. These focus on strengthening the optimization formulations presented, improving the rule-based methods, and incorporating greater realism to improve the argument that such models be used for designing better capacity allocations in operational systems.

Given the increasing congestion and complexity of the air traffic system, the first area of continuing work lies in reformulating or strengthening the formulation of the optimization models in each chapter. The consideration different modeling assumptions, including capacitated time periods in place of slots, as well as the specification of a maximum delay parameter as used in (Bertsimas and Stock Patterson 1998) may help reduce formulation size. However, these simplifications come at the expense of precision and reduce the ability of the modeler to include complex capacity profiles. In addition, formulation improvements such as those in (Glover and Ball 2010) or computational techniques such as those in (Rios and Ross 2008) have the potential to improve solution times and allow for the solution of larger case studies.

In addition, the rule-based approaches specified in §2 have the potential to provide even more powerful tools than any of the optimization formulations. However, they require additional refinement to realize their potential utility. In addition, a modified version of the final delay priority rule may provide a very useful mathematical result in demonstrating a heuristic approach that very closely approximates the results of an optimization model.

Finally, because the models in this dissertation address such a practical problem, they and their derivative principles have the potential to contribute to improving efficiencies in the air traffic system. To this end, an in-depth analysis of them with respect to their interactions with the practical and operational nuances of this system should be undertaken. In this way, they may be better integrated with the body of research and development efforts currently underway to modernize the air traffic system.

References

Aviation Weather Center. *Collaborative Convective Forecast Product: Product Description Document.* Kansas City, Missouri: National Weather Service, 2005.

Balakrishnan, Hamsa, and Bala Chandran. "Algorithms for scheduling runway operations under constrained position shifting." *Operations Research*, 2010.

Ball, Michael O, Robert L Hoffman, Amedeo R Odoni, and Ryan Rifkin. "A stochastic integer program with dual network structure and its application to the ground-holding problem." *Operations Research* 51, no. 1 (2003): 167-171.

Ball, Michael O, Robert L Hoffman, and Avijit Mukherjee. "Ground delay program planning under uncertainty based on the ration-by-distance principle." *Transportation Science*, 2009.

Benjamin, Stanley G, et al. "An hourly assimilation-forecast cycle: The RUC." *Monthly Weather Review* 132 (2004): 495-518.

Benjamin, Stanley G, Georg A Grell, John M Brown, and Tatiana G Smirnova. "Mesoscale weather prediction with the RUC hybrid isentropic-terrain-following coordinate model." *Monthly Weather Review* 132 (2004): 473-494.

Bertsimas, D, G Lulli, and Amedeo R Odoni. "The air traffic flow management problem: an integer optimization approach." *13th International Conference, IPCO 2008.* Bertinoro, Italy: Springer, 2008. 34-46.

Bertsimas, Dimitris J, and Sarah Stock Patterson. "The air traffic flow management problem with enroute capacities." *Operations Research* 46, no. 3 (1998): 406-422.

Bertsimas, Dimitris J, Guglielmo Lulli, and Amedeo R Odoni. "The air traffic flow management problem: an integer optimization approach." *13th International Conference, IPCO 2008.* Bertinoro, Italy: Springer, 2008. 34-46.

Bertsimas, Dmitiris J, and Sarah Stock Patterson. "The traffic flow management rerouting problem in air traffic control: a dynamic network flow approach." *Transportation Science* 34, no. 3 (2000): 239-255.

Brennan, Michael. "Airspace flow programs - a fast path to deployment." *Journal of Air Traffic Control* 49, no. 1 (2007): 51-55.

Buxi, Gurkaran S, and Mark Hansen. "Generating day-of-operation probabilistic capacity profiles from weather forecasts." *Proceedings of 4th International Conference on Research in Air Transportation.* Budapest, 2010. 305-312.

Charba, Jerome P, and Liang Feng. "Automated two hour thunderstorm guidance forecasts." *Proceedings of Conference on Meteorological Applications of Lightning Data.* San Diego, CA, 2005.

Chen, Yudong. "A Dynamic Stochastic Model for the Air Traffic Flow Management Problem." *INFORMS 2009 Conference.* San Diego, California, 2009.

Churchill, Andrew M. "Determining the Number of Slots to Submit to a Market Mechanism at a Single Airport." MS Thesis, University of Maryland, 2007.

Churchill, Andrew M, David J Lovell, and Michael O Ball. "Evaluating a new formulation for large-scale traffic flow management." *Proceedings of the 8th USA/Europe Air Traffic Management R&D Seminar.* Napa, California, 2009.

Cook, Lara S, and Brian Wood. "A model for determining ground delay program parameters using a probabilistic forecast of stratus clearing." *Proceedings of the 8th USA/Europe Air Traffic Management R&D Seminar.* Napa, California, 2009.

Dear, R G. *The dynamic scheduling of aircraft in the near terminal area.* Cambridge, Massachusetts: MIT Flight Transporation Laboratory Report R76-9, 1976.

Fearing, Douglas, Cynthia Barnhart, Dimitris Bertsimas, and Constantine Caramanis. "Equitable and Efficient Coordination of Traffic Flow Management Programs." Working paper, 2009.

Federal Aviation Administration. *Order JO 7110.65S: Air traffic control.* U.S. Department of Transportation, 2008.

Ganji, Moein, David J Lovell, Michael O Ball, and Alex Nguyen. "Resource allocation in flow-constrained areas with stochastic termination times." *Transportation Research Record: Journal of the Transportation Research Board* 2106 (2009): 90-99.

Ghirardelli, Judy E. "An overview of the redeveloped localized aviation MOS program (LAMP) for short-range forecasting." *Proceedings of 21st Conference on Weather Analysis and Forecasting.* Washington, DC, 2005.

Glover, Charles N, and Michael O Ball. "Stochastic integer programming models for ground delay programs with weather uncertainty." *Proceedings of 4th International Conference on Research in Air Transportation.* Budapest, 2010. 217-223.

Helme, M. "Reducing air traffic delay in a space-time network." *IEEE International Conference on Systems, Man and Cybernetics.* Chicago, Illinois, 1992. 236-242.

Hoffman, Robert. "Integer Programming Models for Ground-Holding in Air Traffic Flow Management." PhD Dissertation, University of Maryland College Park, 1997.

Hoffman, Robert, and Michael O Ball. "A comparison of formulations for the single airport ground holding problem with banking constraints." *Operations Research* 48, no. 4 (2000): 579-590.

Kotnyek, Balázs, and Octavio Richetta. "Equitable Models for the Stochastic Ground-Holding Problem Under Collaborative Decision Making." *Transportation Science* 40, no. 2 (2006): 133-146.

Krozel, Jimmy, Ray Jakobovits, and Steve Penny. "An algorithmic approach for airspace flow programs." *Air Traffic Control Quarterly* 14, no. 3 (2006): 203-230.

Liu, P Barry, Mark Hansen, and Avijit Mukherjee. "Scenario-based Air Traffic Flow Management: From Theory to Practice." *Transportation Research Part B* 42 (2008): 685-702.

Lulli, Guglielmo, and Amedeo R Odoni. "The european air traffic flow management problem." *Transportation Science* 41, no. 4 (2007): 431-443.

Manley, Benji. "Minimizing the Pain in Air Transportation: Analysis of Performance and Equity in Ground Delay Programs." PhD Dissertation, George Mason University, 2008.

Metron Aviation. "Updated Operational Concept for "System Enhancements for Versatile Electronic Negotiation"." 2009.

Mukherjee, Avijit, and Mark Hansen. "A dynamic stochastic model for the single airport ground holding problem." *Transportation Science* 41, no. 4 (2007): 444-456.

Myers, Tim, and David Kierstead. "Network Model to Address Capacity/Demand Imbalances in the National Airspace System." *AIAA Guidance, Navigation and Control Conference and Exhibit.* Honolulu, Hawaii, 2008.

Odoni, Amedeo R. "The flow management problem in air traffic control." In *Flow Control of Congested Networks*, edited by Amedeo R Odoni and G Szego. Berlin: Springer-Verlag, 1987.

Richetta, O, and A R Odoni. "Solving Optimally the Static Ground-Holding Policy Problem in Air Traffic Control." *Transportation Science* 27 (1993): 228-238.

Rios, Joseph L, and Jason Lohn. "A comparison of approaches for national traffic flow management." *Proceedings of the AIAA Guidance, Navigation, and Control Conference.* Chicago, Illinois, 2009.

Rios, Joseph L, and Kevin Ross. "Solving high-fidelity, large-scale traffic flow management problems in reduced time." *Proceedings of The 8th AIAA*

Aviation Technology, Integration, and Operations Conference. Anchorage, Alaska, 2008.

Sherali, Hanif D, Raymond W Staats, and Antonio A Trani. "An Airspace Planning and Collaborative Decision–Making Model: Part I—Probabilistic Conflicts, Workload, and Equity Considerations." *Transportation Science* 37, no. 4 (2003): 434-456.

Sherali, Hanif D, Raymond W Staats, and Antonio A Trani. "An Airspace-Planning and Collaborative Decision-Making Model: Part II—Cost Model, Data Considerations, and Computations." *Transportation Science* 40, no. 2 (2006): 147-164.

Sridhar, Banavar, Tarun Soni, Kapil Sheth, and Gano Chatterji. "An Aggregate Flow Model for Air Traffic Management." *Proceedings of the AIAA Guidance, Navigation, and Control Conference.* Providence, Rhode Island, 2004.

Storm Prediction Center. *NWS Product Definition Document (PDD) for: Refinement of SPC Experimental Enhanced Resolution Thunderstorm Outlook.* Norman, Oklahoma: National Weather Service, 2009.

Sun, Dengfeng, and Alexandre M Bayen. "Multicommodity Eulerian–Lagrangian Large-Capacity Cell Transmission Model for En Route Traffic." *Journal of Guidance, Control, and Dynamics* 31, no. 3 (2008): 616-628.

Terrab, Mostafa, and Amedeo R Odoni. "Strategic flow management for air traffic control." *Operations Research* 41, no. 1 (1993): 138-152.

Vossen, Thomas W.M., and Michael O Ball. "Slot Trading Opportunities in Collaborative Ground Delay Programs." *Transportation Science* 40, no. 1 (2006): 29-43.

Vranas, Peter B, Dimitris J Bertsimas, and Amedeo R Odoni. "Dynamic ground-holding policies for a network of airports." *Transportation Science* 28, no. 4 (1994): 275-291.

Vranas, Peter B, Dimitris J Bertsimas, and Amedeo R Odoni. "The multi-airport ground-holding problem in air traffic control." *Operations Research* 42 (1994): 249-261.

Wambsganss, Michael. "Collaborative decision making through dynamic information transfer." *Air Traffic Control Quarterly* 4, no. 2 (1997): 109-125.

Weiss, Mitchell, and Judy E Ghirardelli. "A Summary of Ceiling Height and Total Sky Cover Short-Term Statistical Forecasts in the Localized aviation MOS Program (LAMP)." *Proceedings of 21st Conference on Weather Analysis and Forecasting.* Washington, DC, 2005.

Weygandt, Stephen S, Stanley G Benjamin, Tatiana G Smirnova, John M Brown, and Kevin Brundage. "Hourly convective probability forecasts and experimental high-resolution predictions based on the radar reflectivity assimilating RUC model." *Proceedings of the 13th Conference on Aviation, Range and Aerospace Meteorology.* New Orleans, Louisiana, 2008.

Weygandt, Stephen, and Stanley G Benjamin. "RUC model-based convective probability forecasts." *Proceedings of the 11th Conference on Aviation, Range, and Aerospace Meteorology.* Hyannis, Massachusetts, 2004.

Wiedenfeld, Jerry R. "Localized aviation MOS Program (LAMP): Statistical Guidance of Wind Speed, Direction, and Gusts for Aviation Weather." *Proceedings of 21st Conference on Weather Analysis and Forecasting.* Washington, DC, 2005.