ABSTRACT

Title of Dissertation:	FLIGHT ARRIVAL SCHEDULING MODELS FOR INCORPORATING COLLABORATIVE DECISION-MAKING CONCEPTS INTO TIME-BASED FLOW MANAGEMENT
	Yeming Hao, Doctor of Philosophy, 2021
Dissertation directed by:	David J. Lovell, Professor, Department of Civil & Environmental Engineering; Institute for Systems Research

Time-based air traffic flow management balances demand versus capacity at facilities by assigning Controlled Times of Arrival (CTAs) to incoming flights. However, existing systems do not differentiate delay costs among different aircraft and do not take user preferences into consideration. From a business perspective, it is essential to understand user preferences and to allow users to engage in decision-making.

This dissertation presents results of simulations of strategies to incorporate business-driven airline preferences into these air traffic flow management systems following a Collaborative Decision-Making paradigm. We evaluate optimization models and heuristics to assign CTAs based on user-provided information and priority preferences in a way that minimizes the total CTA delay cost. Potential savings were quantified by comparing the results with the default first-come-first-served (FCFS)

scheme. Monte Carlo simulations are conducted using historical flights data under a variety of realistic scenarios. Results show that our proposed heuristic, 2OptSwap, could reduce CTA delay cost between 20% and 30% relative to the FCFS baseline scheme, maintaining an average of 5.9% gap compared to the optimal solution. It is also shown that the heuristic could potentially realize the same level of cost savings regardless of whether the incoming flights are behind their schedules or not. Additional findings include that by starting the flow management further away, we could achieve more delay cost savings.

The environment in the air space (such as the wind and the facility capacity) is constantly changing. Therefore, a rolling horizon approach was integrated into this air traffic flow management system. This approach allows the system to incorporate the most recent information at each epoch and solve the problem in a dynamic fashion. Real-time fairness metrics and adjustment methods are defined such that performance measurements in the previous epochs can be used for adjustments in future decision-making. Simulation results show that these fairness adjustments can help achieve a fairer benefits distribution among carriers and achieve a win-win solution.

FLIGHT ARRIVAL SCHEDULING MODELS FOR INCORPORATING COLLABORATIVE DECISION-MAKING CONCEPTS INTO TIME-BASED FLOW MANAGEMENT

by

Yeming Hao

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
2021

Advisory Committee:
Professor David J. Lovell, Chair
Professor Michael O. Ball, Dean's Representative
Professor Paul Schonfeld
Professor Mark A. Austin
Dr. Sergio Torres

Dedication

To my family.

Acknowledgements

First of all, I am forever grateful for my advisor, Professor David J. Lovell, for his support and guidance. He welcomed me to his lab even though I had no experience working on the aviation field and secured funding for me so I could focus on learning. He helped me get started quickly by giving me continuous feedback and guidance. He has everything a mentee could ever hope for in a mentor: wisdom, insight, dedication, and supportiveness. Here I want to quote something I once read online: "The advice I continue giving folks looking into PhD programs is to look for an advisor who is kind … Look for the person who is kind, who mentors you, lifts you up. And pro-tip: kind folks are usually the most brilliant." – Dr. Jorge J Jodriguez V. I am truly grateful that I get to work with an advisor exactly matches this description. Professor Lovell genuinely cares about my growth and development as a student and a young scholar. Moreover, he is a friend that cares so deeply about me as a person: he is always supportive when I encounter unexpected challenges. He lifted me up.

I would also like to thank Professor Michael O. Ball for guiding my research and supporting me on his projects. He selflessly takes time, even after retirement, to review my work and give me guidance and feedback. He is a brilliant pioneer in the aviation operations research field and his keen insights always help me stay on the right track. Every time I read his papers, I grew greater respect towards his contributions and deeper gratitude towards his guidance.

I would also like to acknowledge Dr. Sergio Torres at Leidos, Inc. This dissertation topic originated from a funded project between UMD and Leidos. Dr. Torres contributed a lot of brilliant ideas with his rich work experience. Knowing I was interested in extending the topic into long-term research, Dr. Torres remained after the project ended and selflessly kept contributing feedback and, in the end, took the responsibility of being on my committee. This project would not be where it is today without his valuable advice.

I want to take this opportunity to thank Professor Paul Schonfeld as well. I am lucky to have him at every step throughout my graduate school journey. He was there teaching my knowledge, at the start, when I took his class in my first semester; He was there making us feel at home, when he hosted welcome and reunion parties for basically the whole department; he was there connecting us to industry organizations when I worked for the ITS-ITE student chapter; he was there patiently giving me feedback and raising questions, when I defended my Master's thesis; he was there understanding and supporting my decisions when I switched labs and research topics; and here he is, at the end, witnessing my growth and providing advice by being on my PhD dissertation committee. I am grateful that I get to connect with such a kind, wise, and highly respected person and I am grateful to have his name written on the list of my advisory committee and this life chapter.

My sincere thanks also go to Professor Mark A. Austin. His class is always full of lively discussions, his office hour is always intriguing, and his comments on my research are always inspiring. I thank him for all the knowledge he taught, all the

stimulating discussions he led, and all the valuable advice he provided as an indispensable member of my committee.

I also owe a debt of gratitude to my family: my parents, my elder sister, and especially my twin sister and best friend since before birth, Dr. Shuming Hao, for supporting me throughout the writing of this dissertation and my life in general. Grew up in a small rural town and being the first-generation high school student, I never thought studying abroad and gaining a doctoral degree was an option until Shuming convinced me so. If it were not for her, my graduate school journey would never have started.

Special thanks to my boyfriend and soul mate, Lautaro D. Cilenti, for all the love and support. He helped me see a bigger world and coated another layer of color on my life.

Last but not least, I thank all my friends, especially Zhongxiang Wang, Qian Wang, Wenmo Sun, Zixuan Wu, Luanjiao Hu, Liang Liang, Mengshi Li, Chenyang Fang, Danae Mitkas, Sachraa G. Borjigin and many others for their enjoyable company during this unforgettable journey.

Financial support for this research received from Leidos, Inc. is gratefully acknowledged.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AAR	Airport Acceptance Rate
ASPM	Aviation System Performance Metrics
ATA	Actual Time of Arrival
ATC	Air Traffic Control
ATL	Hartsfield-Jackson Atlanta International Airport
CDF	Cumulative Distribution Function
CDM	Collaborative Decision-Making
CSP	Constrained Position Shifting
CTA	Controlled Time of Arrival
EDCT	Expect Departure Clearance Time
ETA	Estimated Time of Arrival
FAA	Federal Aviation Administration
FADE	FAA/Airline Data Exchange, renamed to CDM later on
FCFS	First-Come First-Served
FH	Freeze Horizon
GDP	Ground Delay Program
GDP-E	Ground Delay Program Enhancements
GS	Ground Stop
ICAO	International Civil Aviation Organization
MAD	Mean Absolute Deviation
NSS	Neighbor Searching Size
RBS	Ration by Schedule
RH	Rolling Horizon
SA	Simulated Annealing
STA	Scheduled Time of Arrival
TAS	True Air Speed
TBFM	Time-Based Flow Management
TMI	Traffic Management Initiatives
TRACON	Terminal Radar Approach Control

Chapter 1: Introduction

1.1 Research Background and Problem Statement

In air traffic management, aircraft can be managed in a time-based manner in the en route and terminal environments such that aircraft maintain a certain flow rate that is consistent with the downstream facilities' capacity. The benefits of such a time-based aircraft flow management system include minimizing coordination, reducing the need for vectoring/holding, and efficiently utilizing airport and airspace capacity [1].

In the United States, the Federal Aviation Administration (FAA) has been using a foundational decision support tool, referred to as Time-Based Flow Management (TBFM), at certain facilities for such an aircraft flow management purpose. TBFM enables aircraft to maintain a given spacing behind a preceding aircraft, improving capacity and flight efficiency [1, 2]. It can be used for aircraft arriving to an airport so flights arrive in a time-ordered sequence that is consistent with airport capacity and with adjusted inter-arrival spacing such that landing services can be offered safely and efficiently.

An example of this process is shown in Figure 1. Terminal Radar Approach Control (TRACON) is an FAA Air Traffic Control (ATC) facility within the terminal airspace, using radar and air/ground communications to provide approach control services to aircraft [3]. A metering fix is a location along an established route over

which aircraft are metered prior to entering this terminal airspace [4]. When flights cross the imaginary line called the freeze horizon (FH), they will be assigned controlled times of arrival (CTAs) to the metering fix at the airport. The flights then operate to meet these CTAs.

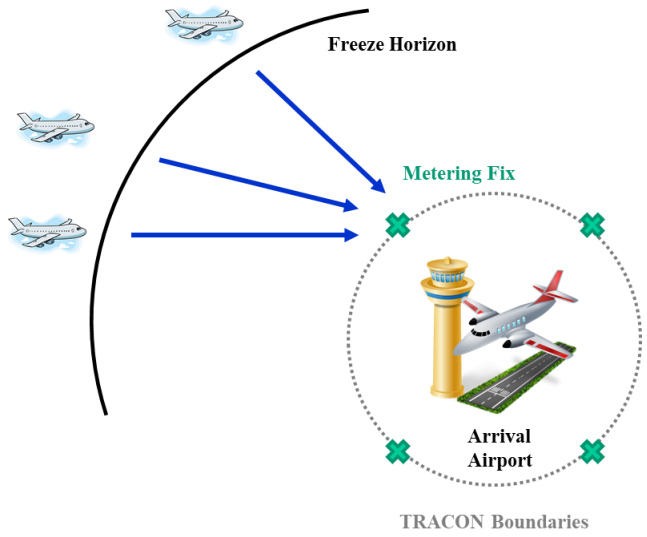

Figure 1 CTA Assignment at the Freeze Horizon in TBFM

As of November 30, 2017, the FAA has deployed TBFM in 28 core airports and 13 non-core airports [2]. The TBFM system currently in use adopts a first-come-first-served (FCFS) protocol to determine the CTAs for flights at various points in the national airspace system. This CTA assignment is based solely on the flights' Estimated Times of Arrival (ETAs) and does not take user preferences into consideration.

However, for carriers, the costs of delays on different flights could be very different due to factors such as aircraft types, and the number of connecting passengers. They might prefer to delay one flight over another to save the overall cost. Additionally,

flights' scheduled times of arrivals (STAs) should be considered in addition to ETAs, since the former is directly related to the business performance metric carriers care about the most. The goal of this dissertation is to investigate opportunities for borrowing Collaborative Decision-Making (CDM) concepts from other air traffic management regimes and adapting them as a means of incorporating these carrier preferences into the CTA assignment in the TBFM process, and more broadly, into any flow management service with the same general purpose as TBFM.

1.2 Contributions

The main contributions of this dissertation can be summarized as follows:

- We offer a detailed simulation scheme that comes with different CTA assignment schemes to reduce the total CTA delay cost relative to STAs. The scheme can be easily adapted to other purposes and other goals, such as a different term or a combination of several terms in the objective function.
- A traditional delay in minutes metric was converted to a delay cost metric with customized flights costs. As stated in the above section, this is directly associated to carrier's business objectives and shows the stakeholders how much our scheme is financially beneficial.
- CDM concepts were incorporated into TBFM. CDM concepts were originally designed to work in the long look-ahead strategic domain, whereas the arrival management in TBFM was designed for the closer tactical domain. This

research applied the CDM concepts in TBFM. In this process, carriers have access to an interface with the air navigation service provider (such as the FAA), which allows them to provide information including their preferences over different flights and participate in the process. This should be beneficial to them because the CTAs assigned will be customized to fit their preferences and interests.

- Our research paves the path for the integration between strategic tools such as ground stops (GSs) and ground delay programs (GDPs) and tactical tools such as TBFM. By assigning CTAs early on, we could control and transfer some delay from terminal areas, and this could lead to better decision-making in those strategical tools.
- The dynamic scheme proposed in the study allows the use of real-time metrics. Such settings allow the system to adjust decision-making based on the historical feedback and can better reach the overall goals.

1.3 Structure of Dissertation

The organization of this dissertation is as follows. In Chapter 2, we summarize the previous work in the related fields. In Chapter 3, methodologies for the CTA assignment problem were introduced and proposed, including the FCFS baseline model, an optimization model that provides the optimal solution, and two heuristic approaches that solve the problem in a faster and more efficient fashion by sacrificing

optimality. In Chapter 4, the simulation system structure and simulation procedures were described. In Chapter 5, simulation results were presented using historical data from Hartsfield-Jackson Atlanta International Airport (ATL) in various settings. In Chapter 6, we introduce the idea of using the rolling horizon approach and real time metric to form a dynamic scheme for the same problem. In Chapter 7, we provide the simulation results for the proposed dynamic scheme. In Chapter 8, we summarize the main conclusions and discuss possibilities for future research.

Chapter 2: Literature Review

This chapter describes the related literature in CDM and the aircraft landing problem and identifies gaps in the research and state-of-the-art system capability.

2.1 Collaborative Decision-Making

CDM evolved from the FADE (the FAA/Airline Data Exchange) program, a program that began in 1993, aiming to explore if the provision of up-to-date airline schedule information to the FAA would result in improved traffic management decision-making [5, 6]. Prior to these developments, air carriers would frequently plan to cancel or delay their flights, without sharing this information with the FAA. The carriers chose not to share the information in cases where they felt only their competitors could benefit from the information, and this hampered system efficiency and throughput. The FADE experiments proved that real-time information sharing could improve air traffic management decision-making and the concept of CDM was thus conceived.

CDM embodies a philosophy for better managing air traffic through information exchange, procedural improvements, tool development, and common situational awareness [7]. Fundamental principles of CDM include rich information exchange among all relevant parties and the use of resource allocation mechanisms that encourage truthful and prompt information sharing. Further, the decision and control

architecture should enable air carriers to make those decisions that involve only their own resources, should seek to treat all operators equitably and, to the extent possible, should infuse carrier preference information within any FAA allocation processes.

Researchers have been studying the positive impact CDM can bring to air traffic management, especially to the planning and control of GDPs. Wambsganss (1996) developed a compression algorithm that uses information exchange between the FAA and carriers to better utilize vacant slots in GDPs. Four scenarios were tested to show the algorithm reduces scheduled delay [5]. Ball et al. (2000) showed that CDM can provide Ground Delay Program Enhancements (GDP-E) in terms of the quality of information, delay, and so on. The analysis was based on CDM implementation data at San Francisco (SFO) and Newark (EWR) airports [7]. Carriers' inputs were considered in many CDM studies. Gilbo et al. (2000) proposed an arrival/departure optimization model that can take into account users' priorities in their traffic demand when allocating airport capacities under GDP-E [8]. Chang et al. (2001) introduced the Flight Schedule Monitor, an interface adopted by 33 FAA facilities at that time that enables easy information exchange in GDPs. The interface allows carriers to provide their inputs, such as carrier schedules/plans and carrier assessments of the certainty of the weather forecast, and allows the FAA to collaborate with carriers and factor in these inputs [9].

Some researchers have extended these CDM concept to fields other than the traditionally applied GDP field. Sheth & Gutierrez-Nolasco (2008) and Rios et al. (2010) incorporated users' preferences and intent during the pre-departure route filing

process. The users are assigned a fixed number of credits at the beginning, and they can allocate the credits among their flights to indicate importance. Notwithstanding that incorporating user preferences will increase delays proportionally to increased user satisfaction, results show that user preferences can be accommodated without inordinately jeopardizing the overall system performance [10, 11].

Vlachou and Lovell (2013) modeled a compact vocabulary for carrier priorities and acceptable flight delays that could be incorporated in a fair air resource allocation procedure such as that described by Vakili (2009) [12]. The ration by schedule (RBS) mechanism was used as the baseline since it was the practice in use for airspace flow programs. The delay and weighted delay of the two proposed allocation mechanisms were constantly lower than RBS and RBS with substitutions [13].

Raj (2018) modeled the use of a similar vocabulary in the Collective Trajectory Options Program (CTOP), with additional controls on the vocabulary to prevent strategic gaming. Priorities of flights by the carrier were incorporated, aiming at not merely reducing an individual carrier's delay but also the total delay in the system [14].

A more recent paper by Idris et al. (2019) has analyzed modifying the TBFM allocation algorithm based on an accrued delay prioritization scheme [15]. While this approach does not explicitly employ user preference information, it does lead to more equitable solutions, a key CDM principle.

Xu et al. (2020) proposed a framework that incorporated the CDM concepts into trajectory options and pre-tactical delay management by allowing airspace users to initially indicate their preferred trajectories [16].

Given that CDM problems involve complicated exchanges between carriers, various methodological approaches have been adopted to solve these problems, particularly framing these exchanges as discrete optimization problems. In particular, (mixed) integer programming (MIP/IP) models are useful approaches to finding optimal solutions [16]. Relaxation or heuristic approaches are usually adopted for real time solutions [17-20].

Despite all the above researchers' efforts in demonstrating the benefits of CDM in these other fields, most of their models have not yet been applied into the real world, and to the best of the author's knowledge, there is still no existing research applying these CDM concepts in the dynamic tactical time frame, such as scheduling en route flights to the destination airport. Our study aims to bridge this gap.

2.2 Aircraft Landing Problem

This section summarizes research that focused on the aircraft landings problem (ALP) at airports since it has overlapping goals and interests with our study. The aircraft landing problem (ALP) consists of scheduling the landing of aircraft onto runways in an airport by assigning to each aircraft a landing time and a specific runway while respecting different operational constraints [21]. This problem is also referred to as

the aircraft sequencing problem and it is generally confined to the terminal area [22, 23].

A widely used concept in ALP is called Constrained Position Shifting (CSP), a methodology proposed by Dear and Sherif (1989) for sequencing and scheduling of aircraft in high density terminal areas [24]. The method allows aircraft to be shifted in the landing sequence relative to the FCFS sequence. A maximum deviation number is set so that no aircraft will be shifted more than the allowed range. For example, if CPS = 2, it means each flight can shift at most two positions relative to its original FCFS position in the sequence. The concept was later adopted and further developed in various research [25, 26]. However, flights come in different conditions and treating them uniformly (with the same position CPS limit) may not be ideal for the complexities in the real world. Mesgarpour et al. (2010) realized this issue and proposed using time shifting instead of position shifting [27]. Time shifting limits the deviation of landing time of an aircraft from that obtained using the FCFS sequence, where the limit can be dependent on aircraft type.

Beasley et al. (2000) proposed a mixed integer formulation, its linear programming (LP) relaxations, and a heuristic to solve the ALP problem [28]. The computational results were presented for test problems involving up to 50 planes and 4 runways for both the optimal and the heuristic solution. The problem was scoped to when the airplanes enter within the radar range of ATC and require the ATC to assign them landing times. The paper assumes that cost is incurred only if the airplane is required to fly at a speed other than the cruise speed. The paper further assumes the cost

function is composed of two linear portions and thus the problem was able to be formulated with a linear objective function. However, from an airline's perspective, the cost associated with the speed adjustment is not the only cost; for example, they care about the cost associated with their flights' on-time performance, in other words, the delay cost.

Balakrishnan and Chandran (2006) presented an approach for minimizing the makespan (i.e., landing time of the last aircraft in the sequence) in the presence of CPS for a single runway and have demonstrated its effectiveness in a real-world setting at Denver International Airport [29]. Again, the approach was only limited to TRACONs, instead of further en-route traffic management. The latest possible arrival time was set at 60 minutes after the ETA, implying that they could put an aircraft on hold for at most an hour. However, terminal airspace holding of an hour is highly inefficient and such a large bound could discourage carriers from participating in the assignment. Additionally, no equality among different carriers was studied. The authors later (2010) added arbitrary aircraft-dependent cost functions (by using discrete-time models) but did not implement them [25].

Lee and Balakrishnan (2008) extended the previous work frame by minimizing the total delay costs where the individual delays were calculated as the differences between the actual and estimated arrival times [30]. In this scheme, the sum of estimated arrival times is constant; therefore, the problem of minimizing the sum of arrival times is equivalent to that of minimizing the average delay relative to ETAs. This is reasonable in the terminal area for saving fuel, but for a problem of larger

scope, airlines also care about flights' delay relative to the STAs since this will be their on-time performance measure. The analysis suggested that a time advance of up to 3 min is optimal in most practical scenarios. The individual aircraft were not treated differently on this issue. The study also pointed out that in the worst-case scenarios, throughput maximization could result in up to a 20% increase in the fuel cost and up to a 9% increase in the direct operating costs relative to FCFS, which justifies the benefits of switching from maximizing the throughput to minimizing the delay cost.

As mentioned above, there are various research in the open literature that tackle some version of the ALP problem, but they are mostly focused solely on terminal airspace. However, the reasons behind the problems in the terminal airspace start to develop when the airplanes are still far from the terminal. Additionally, aircraft maneuvers that happen at low altitudes with frequent thrust transients, as would be required in the terminal airspace, lead to significant increases in noise, fuel consumption and emissions [31-33].

Researchers have started exploring this issue in a further en route scope. Jones et al. (2018) proposed integer programming models that allocate flight arrival slots by performing en route speed control to reduce fuel consumption. The results show that if the arrival time was established much earlier, then that delay could be realized by having the aircraft fly slower while still at a higher altitude, which would burn much less fuel than the described maneuvers at the terminal airspace [33].

The ATH Group developed Attila, an en-route, self-metering aircraft time sequencing system that starts managing flights from a further distance and generating solutions around the airlines business goals [34]. Attila has been tested by Delta Air Lines and has demonstrated to deliver significant benefit in an operational environment. but it was an intra-airline arrangement and no other carriers participated in this implementation [35].

In summary, the limited scope of ALP leads to limited improvement scope since the space and time is limited. A more effective way to eliminate terminal delay, reduce fuel consumption and emissions, and reduce terminal area noise is to better manage the arriving flights when they are still further away. Researchers have started exploring this issue, but schemes that handle user preferences and CDM concepts in such a tactical en route air traffic flow management system still need to be developed.

Our study incorporates these ideas and assigns flights CTAs from further away while incorporating user preferences following a CDM paradigm. Under this setting, we will have better control and understanding of terminal airspace usage and performance than the traditional ALP problem setting. Additionally, our approach calculates the feasible CTA range using each aircraft's specific limits, instead of using the same value for all the aircraft as in some of these ALP studies.

2.3 Chapter Conclusion

In this chapter, comprehensive reviews on relevant fields of study were presented. First, the philosophies of CDM and its current applications were analyzed. Secondly, summaries on a relevant problem, ALP were provided. Lastly, we pointed out several research gaps in these fields. As stated in Section 2.1 and 2.2, the current TBFM system does not offer variations that provide flexibility towards carriers' needs and is not combined with other strategic tools.

Given the above reviews in this chapter, there is strong motivation to improve the TBFM system. New tactical flow management schemes that can absorb the beneficial CDM concepts and extend some of the ALP ideas into the en route and terminal management need to be developed.

The goal of this dissertation is to demonstrate the potential benefit of the development of a CDM mechanism that infuses carrier cost and preference data into the TBFM slot assignment process. Specifically, we modify the TBFM allocation algorithm to take into account carrier cost information. In addition to the IP formulations, approaches that are more practical to implement and adapt to any needs in the real world are developed. These approaches take the existing operation schemes (FCFS) as baseline solutions and improve the results towards any desired direction. Improvements are quantified using various performance metrics.

Chapter 3: Methodologies for the CTA Assignment Problem

This chapter presents several methodologies for the CTA assignment problem. We first describe FCFS, the current scheme being used in TBFM. We then propose an optimization model and two heuristics for the same problem.

The data flow of the CTA assignment system is shown in Figure 2. As part of the TBFM system, the CTA assignment system takes information including flights' ETAs and AARs (or more information including carrier preferences, depending on which CTA assignment method is in use) as input, then provide CTA assignments and performance metrics for all flights as output. In the baseline model, FCFS, no carrier preferences are considered. In the proposed optimization model and heuristics, carriers can participate in the CTA assignment process by providing self-defined preferences following the given rules, and the system assigns CTAs based on these preferences. The CTAs generated are passed to ATC to be implemented, and the performance metrics are passed to the TBFM system for evaluation and feedback.

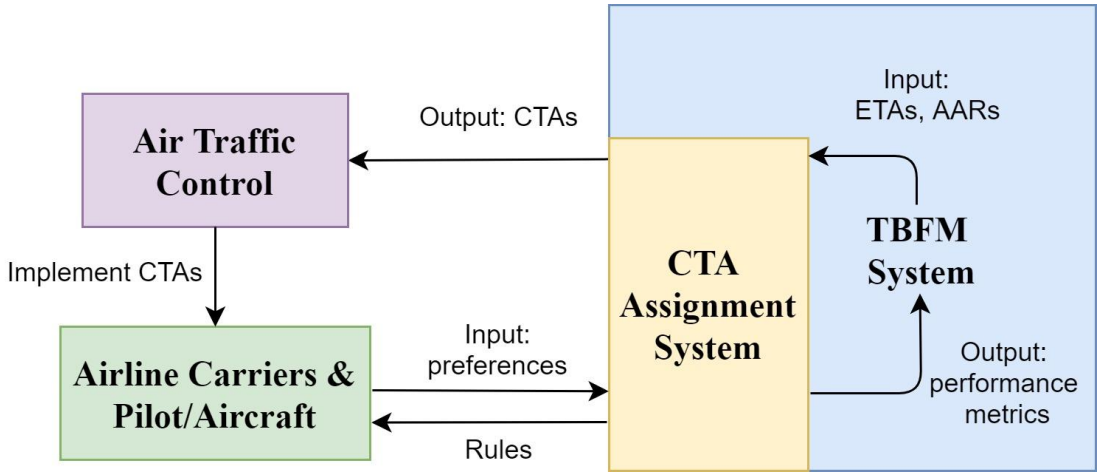

Figure 2 Data Flow of the CTA Assignment System

The basic assumptions of this problem are:

- Carriers are willing to provide flight information in the desired format to be used for the CTA assignment. This includes self-defined priority labels for their flights.
- Carriers are most interested in reducing their total CTA delay cost, the cost incurred when the CTA is later than the STA. The total CTA cost consists of fuel cost, non-connecting passenger delay cost, connecting passenger delay cost, other delay cost, and priority penalty cost.
- All flights follow the CTA assignments and will adjust their operations to meet the CTAs assigned.
- Each flight has its own cost coefficient for delay per minute.
- All flights are pre-assigned to one metering fix. We do not allow the metering fix of a flight to be changed.

3.1 Baseline Model : FCFS Scheme

The FCFS CTA assignment scheme is used in TBFM today and serves as the baseline model for this study. This scheme assigns CTA slots to flights on a predicted order-of-arrival basis and the process is shown in Figure 3. The CTA slot length is calculated using the airport arrival rate (AAR) using Eq.(1).

$$\mathrm{L} = \frac{1}{\mathrm{AAR}} \times 3600 \tag{1}$$

Where AAR is in aircraft per hour, and L is the CTA slot length in seconds.

Each CTA slot represents a CTA that could be assigned to one flight. Given an incoming flight in the list, we first try to assign the slot that corresponds to the flight's ETA. If this slot is occupied or infeasible, we consider advancing the flight to a slot earlier than its ETA, starting with the closest slot available, and moving incrementally up to earlier ones if previous checked slots are not available, up to a limit of the maximum advance time (t^{MAXADV}). Each flight's maximum advance time can be calculated using Eq. (2). In this way, the assigned CTA will not violate the speed range limit for controllers.

$$t^{\mathrm{MAXADV}} = \frac{D_{\mathrm{FH}}}{s_{\mathrm{curr}}} - \frac{D_{\mathrm{FH}}}{s_{\mathrm{max}}} \tag{2}$$

where D_{FH} is the distance between the freeze horizon and the metering fix. For short distance flights with planned distance L shorter than 250 nautical miles (NM), $(L - 50)$ NM was used instead of D_{FH}, where 50 NM is the nominal distance to the top of descent; s_{curr} is the aircraft's current speed; s_{max} is the aircraft's maximum speed allowed.

If an aircraft cannot either be assigned to the slot corresponding to its ETA or be advanced up to the limit of its maximum advance time, we delay the flight by assigning the earliest available CTA slot later than its ETA slot. This is when the delay relative to ETA (referred to as "ETA delay" in this dissertation) occurs. There is no maximum delay limit in this study because the system needs to make sure all the flights get assigned CTAs. Practically, moving a flight to a later slot corresponds to a speed decrease en route, and if this cannot be accommodated, some physical vectoring or holding must occur.

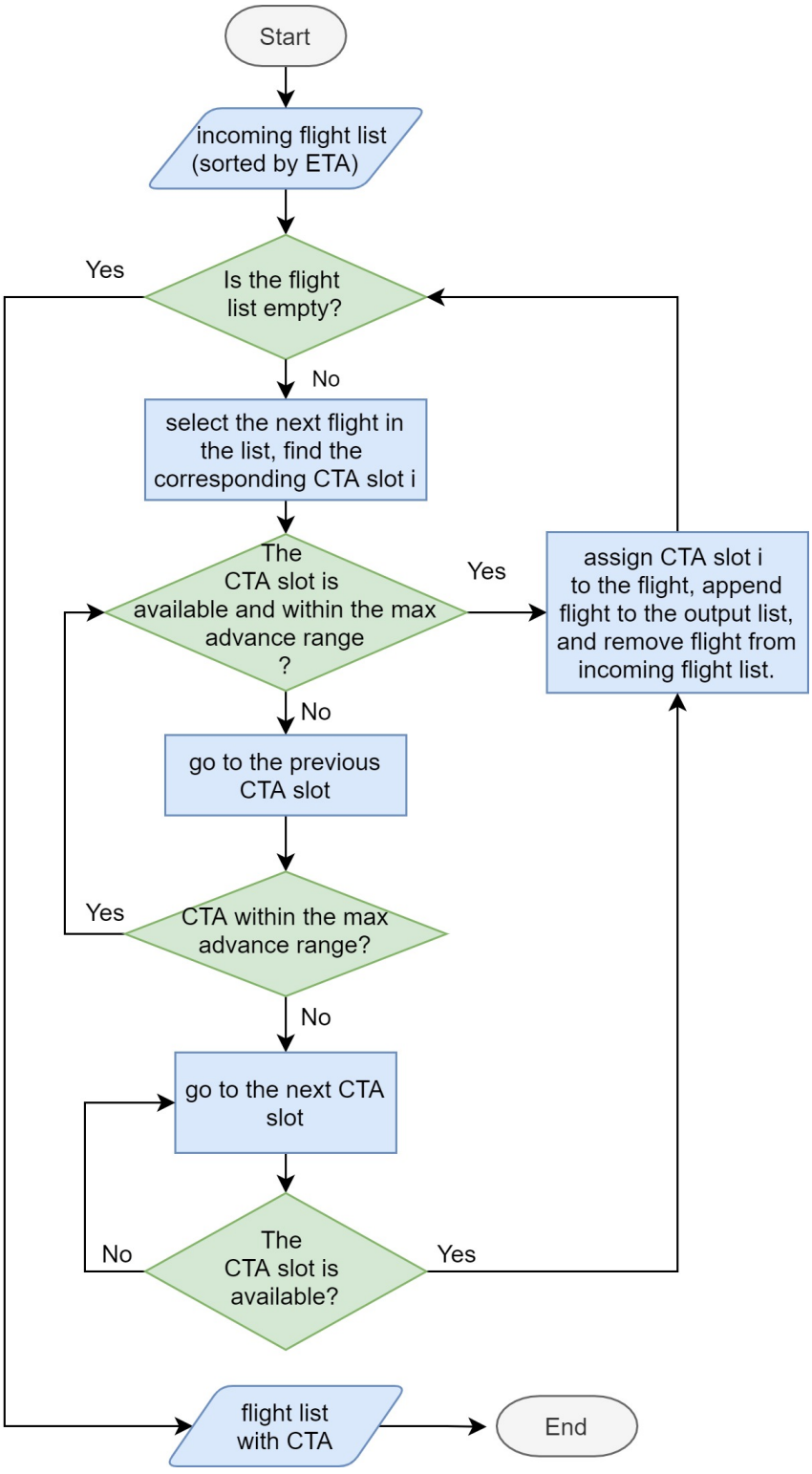

Figure 3 Flow Chart for FCFS

We also provide another variation of this FCFS scheme where advancing a flight is not allowed. That is, if an aircraft cannot be assigned to the slot corresponding to its ETA, we delay the flight by assigning the earliest available CTA slot later than its ETA slot. This scheme is a more realistic representation of the current rules in TBFM in the US today since TBFM does not allow advancing flights. The process is given in Figure 4. To distinguish this scheme from the previously mentioned one, we refer to it as FCFS_w/o_ADV.

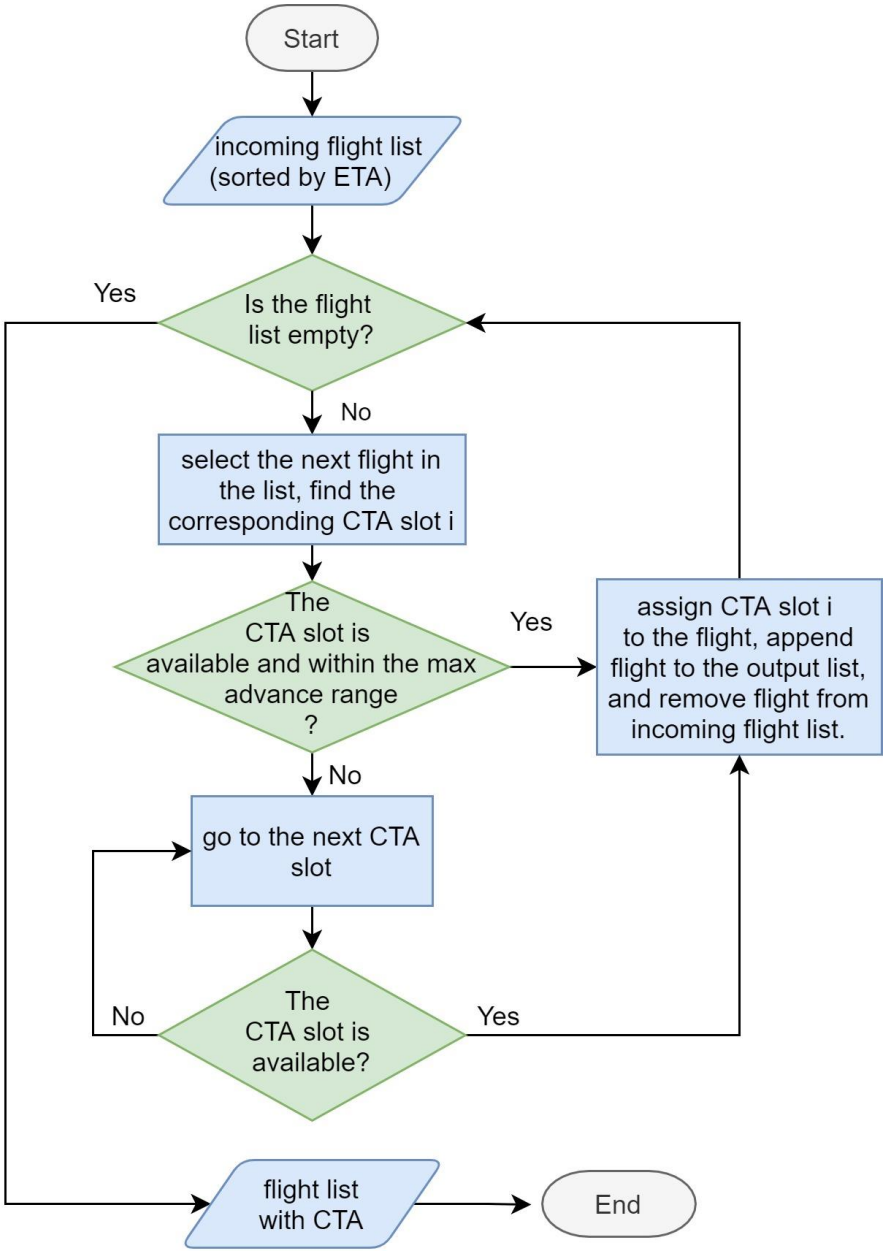

Figure 4 Flow Chart for FCFS_w/o_ADV

3.2: An Optimization Model Using Integer Programming

This chapter presents an optimization model using integer programming for the CTA assignment problem. This formulation aims at minimizing the total CTA delay cost while assigning CTAs to all flights. The CTA delay (relative to STA) occurs when a flight's CTA is later than its STA. The total CTA cost consists of fuel cost, non-connecting passenger delay cost, connecting passenger delay cost, other delay cost, and priority penalty cost.

The fuel cost can be calculated using airplane fuel consumption data but is excluded in our simulations at this moment due to data access. Some existing research have pointed out that fuel cost range stays rather constant within a certain speed range at high altitude, which implies that one could fly at any speed within a range and use nearly the same amount of fuel for a given distance travelled [33]. However, this range should be quantified for each aircraft type and the fuel consumption coefficient in use should capture the differences at different speeds, if applicable.

In our simulations, the non-connecting passenger delay cost and the connecting passenger delay cost can be calculated by multiplying the corresponding cost coefficient by the number of non-connecting passengers and the number of connecting passengers onboard.

Other delay cost consists of all remaining delay costs including connecting crews cost, aircraft turnaround cost, etc.

The priority penalty cost is added to this objective function to allow airlines to customize different priority values for different flights. Examples and rules on how to set the priority values can be found in Section 4.2.4. Under this setting, the model will try to assign less delay to flights with higher priority values, compared to the flights with lower priority values. The details of these delay calculations are described in Section 4.2.3. We also propose alternatives in the same section for when the cost data is not accessible, where a flight cost ranking scheme can be used instead.

3.2.1 Notations

The notations of the model are listed in Table 1.

Table 1 Notations

Sets	Description
$\boldsymbol{i \in F}$	The set of flights in the system
$\boldsymbol{j \in S}$	The set of CTA slots in the system
Decision Variables	
$\boldsymbol{x_{ij}}$	$= \begin{cases} 1, & \text{if flight } i \text{ is assigned to CTA slot } j \\ 0, & \text{otherwise} \end{cases}$
Parameters	
$\boldsymbol{c_i}$	the delay cost coefficient for flight i, \$/min
$\boldsymbol{c_i^{\mathrm{f}}}$	the fuel delay cost coefficient for flight i, \$/min
$\boldsymbol{c^{\mathrm{pax}}}$	the non-connecting passenger delay cost coefficient, \$/min/passenger
$\boldsymbol{n_i^{\mathrm{pax}}}$	the number of non-connecting passengers on flight i
$\boldsymbol{c^{\mathrm{connpax}}}$	the connecting passenger delay cost coefficient, \$/min/passenger
$\boldsymbol{n_i^{\mathrm{connpax}}}$	the number of connecting passengers on flight i
$\boldsymbol{c_i^{\mathrm{o}}}$	other delay cost coefficient for flight i, \$/min
$\boldsymbol{c^{\mathrm{p}}}$	the cost coefficient for priority penalty, \$/min

$\boldsymbol{p_i}$	the priority label for flight i
$\boldsymbol{t_i^{\mathrm{STA}}}$	the STA for flight i
$\boldsymbol{t_i^{\mathrm{ETA}}}$	the ETA for flight i
$\boldsymbol{t_j^{\mathrm{CTA}}}$	the corresponding CTA for slot j
$\boldsymbol{t_i^{\mathrm{CTAFCFS}}}$	the corresponding CTA for flight i in FCFS scheme
$\boldsymbol{t_i^{\mathrm{MAXADV}}}$	the maximum advance time for flight i, mins
$\boldsymbol{t_i^{\mathrm{MAXDLY}}}$	the maximum delay time for flight i, mins
$\boldsymbol{d_{ij}}$	the CTA delay incurred if flight i is assigned for slot j, mins
$\boldsymbol{d_{ij}^{\mathrm{ETA}}}$	the ETA delay incurred if flight i is assigned for slot j, mins
$\boldsymbol{w^{\mathrm{ETA}}}$	the weight of ETA delay cost in the objective function

3.2.2 Model Formulation

The problem can be formulated as follows:

$$\min \sum_{j\in S}\sum_{i\in F}\left(c_i\, x_{ij}\, d_{ij}\right) + w^{\mathrm{ETA}}\sum_{j\in S}\sum_{i\in F}\left(c_i\, x_{ij}\, d_{ij}^{\mathrm{ETA}}\right) \tag{3}$$

Subject to:

$$\sum_i x_{ij} \le 1, \forall j \in S \tag{4}$$

$$\sum_j x_{ij} = 1, \forall i \in F \tag{5}$$

$$x_{ij}(t_i^{\mathrm{ETA}} - t_j^{\mathrm{CTA}}) \le t_i^{\mathrm{MAXADV}}, \forall i \in F, j \in S \tag{6}$$

$$x_{ij}\left(t_j^{\mathrm{CTA}} - t_i^{\mathrm{CTAFCFS}}\right) \le t_i^{\mathrm{MAXDLY}}, \forall i \in F, j \in S \tag{7}$$

$$c_i = c^{\mathrm{pax}} n_i^{\mathrm{pax}} + c^{\mathrm{connpax}} n_i^{\mathrm{connpax}} + c_i^{\mathrm{f}} + c_i^{\mathrm{o}} + c^{\mathrm{p}} p_i, \forall i \in F \tag{8}$$

$$d_{ij} = max\left(t_j^{\mathrm{CTA}} - t_i^{\mathrm{STA}}, 0\right), \forall i \in F, j \in S \tag{9}$$

$$d_{ij}^{\mathrm{ETA}} = max\left(t_j^{\mathrm{CTA}} - t_i^{\mathrm{ETA}}, 0\right), \forall i \in F, j \in S \tag{10}$$

$$w^{ETA} << 1 \tag{11}$$

$$x_{ij} = \{0,1\}, \forall i \in F, j \in S \tag{12}$$

Eq. (3) is the objective function. In addition to minimizing the total CTA delay cost, we added a term of the total ETA delay cost with a much smaller weight (see Eq.

(11)) than the CTA delay cost term such that the model will achieve a smaller ETA delay cost for any solutions that generates the same CTA delay cost. For example, consider a single flight assignment where a flight has an ETA of 9:00 am and a STA of 9:10 am. The model will try to assign it a CTA of 9:00 am, rather than 9:10 am to avoid the additional ETA delay cost, although they generate the same CTA delay cost of 0.

Constraint (4) guarantees that each CTA slot can only be assigned to at most one flight. Constraint (5) guarantees that all flights are assigned to exactly one CTA slot. Constraint (6) is the maximum advance time constraint. This is used to define the feasible CTA range. The value of t_i^{MAXADV} can be calculated based on the aircraft's current speed, allowed speed limits, and the distance of the aircraft from the metering fix using Eq. (2) provided in Chapter 3.1. Constraint (7) is the maximum delay time constraint. The CTA assigned in the optimization model cannot be more than the t_i^{MAXDLY} mins later than the CTA assigned in FCFS for the same flight i. This constraint makes sure no flight is being negatively impacted too much by switching from the FCFS baseline scheme to our optimization model. Constraint (8) is used to calculate the delay cost coefficient for each flight. As discussed in the beginning of Section 3.2, the CTA delay cost consists of fuel cost, non-connecting passenger delay cost, connecting passenger delay cost, other delay cost, and priority penalty delay cost. Constraint (9) provides the calculation for CTA delay in minutes. By computing delay relative to the STA, the cost function reflects operating costs to the airline. Constraint (10) provides the calculation for the ETA delay used in the

second term of the objective function. Constraint (11) guarantees that the weight coefficient of the ETA delay cost is much smaller than the weight of the CTA delay cost in the objective function such that the solution is still driven by the minimizing the CTA delay cost and the ETA delay cost should only matter when the CTA delay cost is the same. The value of this coefficient can be determined in a set of sensitivity analysis. Constraint (12) is the binary constraint for the decision variable.

Additionally, the following constraints could be considered to be added to future versions of optimization models:

- Jet route overtaking constraints

Air traffic controllers cannot typically allow aircraft within a certain distance on the same jet route to overtake each other. If our CTA assignment is conducted when the aircraft are close to the destination airport and the jet routes are very limited, we could add this overtaking constraint to limit these overtaking movements among some flights sharing the same jet route. In order to simulate this completely, one would need real-time trajectory data. At the level of fidelity of this simulation, we do not model this detail. As an alternative for simulation purposes, we could randomly label some flights that are not able to change their position within the CTA sequence.

- Airline's own preference in precedence relations constraints

Airline preferences may dictate that one aircraft should land before another one. It is a practical constraint especially for hub airports where many connecting passengers need to connect to the next flight. For our simulation, we could generate some

precedence relations within each carrier to guarantee some flights' CTAs are always ahead of or later than some other flights' CTAs.

- Constraints for flights that already have an EDCT delay

Expect Departure Clearance Times (EDCT) is the runway release time assigned to aircraft due to Traffic Management Initiatives (TMIs) that require holding aircraft on the ground at the departure airport. In situations where some flights already have EDCT delays and are already behind schedule, constraints could be added to make sure they can only get the same or earlier CTAs in our optimization model, compared to their CTAs assigned in the FCFS, to avoid additional delays. It is worth mentioning that even without adding these constraints, our model still outperforms the FCFS baseline model on avoiding excessive delays for flights with EDCTs since we minimize delays relative to the STAs, instead of ETAs.

3.3 Heuristic Approaches

In Section 3.2, an integer programming model was provided for the CTA assignment problem. The challenge with real-world implementations of such models is that the problem size can grow to the point that the IP cannot be solved quickly enough. Furthermore, it is difficult to have discussions with stakeholders about the various trade-offs that are inherent in the optimization model since they are hidden behind a screen of complicated mathematics. Heuristic algorithms, while not guaranteeing a globally optimal solution, can still find good solutions in reasonable time, and are

more transparent in their behavior. This section presents such heuristic algorithms for the CTA assignment problem.

The heuristic algorithm proposed in this section has two variations: the 2OptSwap and the 2OptSwap_SA, and they are both implemented based on outputs from the FCFS assignment.

3.3.1 The 2OptSwap Scheme

The 2OptSwap CTA assignment scheme is inspired by a local search algorithm first designed to solve the traveling salesman problem [36]. The idea is to improve the results by swapping the position of every possible pair in the solution set (CTA slot assignments) and update the solution set if the swapping makes improvements [37, 38].

Our 2OptSwap starts with a FCFS scheme and adds a swapping step after FCFS to reduce the total CTA delay cost. Each feasible pair of flights is evaluated to see if swapping their CTAs can lead to a lower cost. A set of feasibility checks are conducted to examine if the swapping is allowed: the *maximum advance time* check, the *maximum delay time* check, the *carrier preference* check, and the *CTA delay cost* check.

The *maximum advance time* check examines if the swapping requires the controllers to speed up the flight to where the maximum speed limit is violated. The calculation can be done following Eq. (2) in Section 3.1.

The *maximum delay time* check examines if the swapping generates any CTA delay that exceeds the maximum value, compared to the flight's CTA in the FCFS assignment scheme. We set a fixed value for all flights' maximum delay time, e.g., 10 min, and only allow the swapping if neither of the two flights' CTA gets delayed more than this value, compared to its original CTA in FCFS. The purpose of this check is the same as stated in constraint (7) in the optimization model; it makes sure that no flight is being negatively impacted drastically by switching from the FCFS baseline scheme to our model. Depending on the situations and values, this may require the flights to do path stretching and vectoring to meet the new assigned CTAs.

The *carrier preference* check only allows swapping if a) the two flights are from the same carrier, or b) the two flights are from different carriers, but the flight being swapped back has priority level less than or equal to that of the flight being swapped forward. Each carrier is required to provide priority labels for their flights. One common scheme for priority labels is that they range from 1 (lowest priority) to 5 (highest priority), where each priority label group contains 20% of the flights of that carrier's whole fleet. Some values we recommend airlines to take into consideration when generating their own priority labels can be given, such as the number of connecting passengers within a time range, aircraft types and fuel consumption, numbers of seats and passengers on board, deviations from their STAs, but airlines can remain confidential on how they construct their own priority labels as long as they provide the labels in the desired format and meets all the rules.

The *CTA delay cost* check guarantees that the system only accepts the swapping if it results in reduced CTA delay cost (calculated using Eq.(8) in Section 3.2). This can be easily changed to other interested metrics. More comprehensive cost schemes could also be applied.

Due to swapping feasibilities and calculation time consideration, the system does not conduct these checks between every pair of flights in the dataset. Instead, it conducts these checks for every flight and succeeding flights within the neighbor searching size in the list, ranked by ETAs. A larger neighbor searching size is computationally more expensive but does not necessarily bring a significant amount of additional swaps or CTA delay cost savings, because eventually the swaps start to violate the above time checks. Different values for this neighbor searching size parameter are tested in simulations and the results can be found in Section 5.2.

The 2OptSwap algorithm is given below:

Table 2 Algorithm for 2OptSwap

1. Given flightsTable with FCFS solution
2. Calculate MAX_DLY for each flight and save in flightsTable
3. Do until no improvement is made
3.1 for every flight i in flightsTable except the last one
3.1.1 for the following flights j within the neighbor searching size
3.1.1.1 if swapping the CTA of i and j passes the MAX_ADV, MAX_DLY, and the carrier preference check
a) Δ = the total CTA delay cost change if swapping the CTA of flights i and j
b) if $\Delta < 0$ swap the CTA of i and j
4. Return flightsTable with new CTAs

An example of two flights is presented in Figure 5 and Table 3 to illustrate how these CTA assignment schemes work. Assuming there are two incoming flights. Flight 1 has an ETA of 16:03, a STA of 16:10, and delay cost of $\$c_1$ per minute and Flight 2 has an ETA of 16:04, a STA of 16:05, and delay cost of $\$c_2$ per minute (shown in Figure 5). The available CTA slots at the destination airport are 5-minute windows: 16:05, 16:10, … (shown in Table 3).

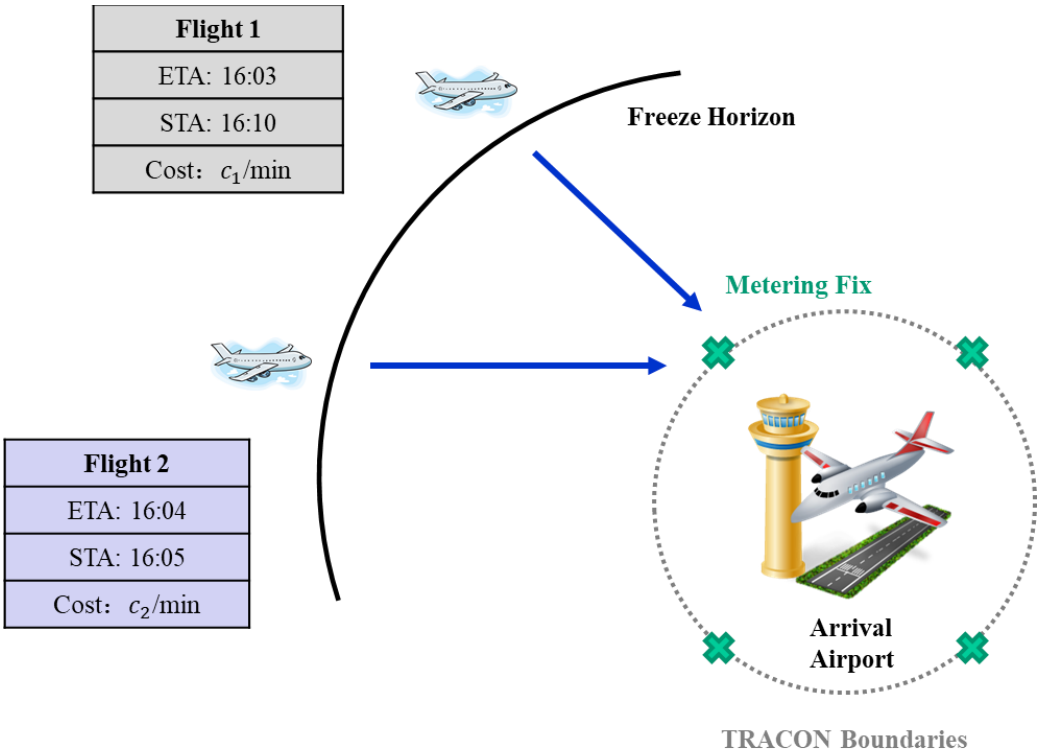

Figure 5 Example of Two Incoming Flights

If the CTA slots are assigned using a FCFS scheme based on the flights' ETAs, Flight 1, which has the earlier ETA, will be assigned a CTA slot of 16:05 and Flight 2 will be assigned a CTA slot of 16:10. However, in this way, Flight 1's CTA is actually

ahead of its STA (16:10), and Flight 2's CTA is behind its STA (16:05) for 5 minutes. The total delay cost for these 2 flights will be $\$5c_2$.

If our 2OptSwap algorithm is applied after FCFS, swapping the FCFS CTAs between these 2 flights allows Flight 2 to take the CTA slot of 16:05, and Flight 1 to take the CTA slot of 16:10. In this case, both flights' CTAs after swapping are ahead of or equal to their STAs. The total delay and the total delay cost are thus 0, which is less than the value in FCFS. Assuming this swapping also passes the three other feasibility checks, the swapping result is then accepted as the final CTA assignment for these 2 flights.

Table 3 Example for 2 Flights: CTA Slots Assignment

CTA slots	Assignment (FCFS)	Assignment (2OptSwap)
….		
16:05	Flight 1 (Delay Cost: \$0)	Flight 2 (Delay Cost: \$0)
16:10	Flight 2 (Delay Cost: $\$5c_2$)	Flight 1 (Delay Cost: \$0)
…		
Total Delay Cost (\$)	$5c_2$	0

3.3.2 The 2OptSwap with Simulated Annealing Scheme

This section provides 2OptSwap_SA, a 2OptSwap CTA assignment scheme combined with simulated annealing (SA) approach.

Simulated annealing is a technique proposed by Kirkpatrick et al. (1983) and independently by Černý (1985) for finding the global minimum of a cost function that

may possess several local minima [39, 40]. Simulated annealing can be combined with local optimization approaches and can help these approaches avoid being trapped in local optima by allowing an occasional uphill move. This is done under the influence of a random number generator and a control parameter called the temperature. The algorithm is given below [41]:

Table 4 Algorithm for Simulated Annealing

1. Get an initial solution S.
2. Get an initial temperature T > 0
3. While not yet *frozen* do the following.
3.1 Perform the following loop L times.
3.1.1 Pick a random neighbor S' of S.
3.1.2 Let $\Delta = cost(S') - cost(S)$.
3.1.3 If $\Delta \leq 0$ (downhill move),
Set $S = S'$.
3.1.4 If $\Delta > 0$ (uphill move),
Set $S = S'$ with probability $e^{-\Delta/T}$.
3.2 Set $T = rT$ (reduce temperature).
4. Return S.

Where "frozen" in step 3 refers to a state in which no further improvement in $cost\ (S)$ seems likely. r is a cooling ratio, $0 < r < 1$.

Using the above logic in the simulated annealing algorithm, we update the algorithm of the 2OptSwap scheme combined with simulated annealing approach (2OptSwap_SA) as below:

Table 5 Algorithm for 2OptSwap_SA

1. Given flightsTable with FCFS solution
2. Calculate MAX_DLY for each flight and save in flightsTable
3. Set initial temperature T_init, stopping temperature T_min, and cooling factor α
4. Do until no improvement is made or reach the stop temperature
4.1 for every flight i in flightsTable except the last one
4.1.1 for the following flights j within the neighbor searching size

4.1.1.1 if swapping the CTA of i and j passes the MAX_ADV, MAX_DLY, and the carrier preference check
a) Δ = the total CTA delay cost change if swapping the CTA of flights i and j
b) if $\Delta < 0$ swap the CTA of i and j
c) else (if $\Delta \geq 0$) swap the CTA of i and j with probability $e^{-\frac{\Delta}{T}}$
4.2 Set T = αT
5. Return flightsTable with new CTAs

The major differences between 2OptSwap and 2OptSwap_SA are highlighted in blue color.

3.4 Chapter Conclusion

We started this chapter by introducing FCFS, the current scheme used in the CTA assignment problem. An IP optimization model and two heuristic approaches, 2OptSwap and 2OptSwap_SA, are then provided. The IP optimization model provides the optimal solutions while the heuristic approaches could offer fast and "transparent" procedures. We can easily track what happens in each step (e.g., which two flights gets swapped and why). The IP optimization model and the two heuristics all need the FCFS solution as an input for the maximum delay time constraint. Simulation experiments on these models are then performed and described in the following chapters.

Chapter 4: Simulation Setup

4.1 Simulation System Structure

The simulation system has multiple layers and many stochastic inputs. It is detailed enough that Monte Carlo simulation is an appropriate tool with which to study design alternatives and consequences. The simulation system structure used for this dissertation was constructed based on a package called Qsim, which was originally developed by Leidos, Inc. in the language of IGOR Pro [42]. Qsim simulates an incoming flights list based on aggregated flight counts, assigns CTAs to flights based on a FCFS basis, and analyzes performances metrics such as delays and flight inter-arrival times. The process and structure of Qsim is given in Figure 6.

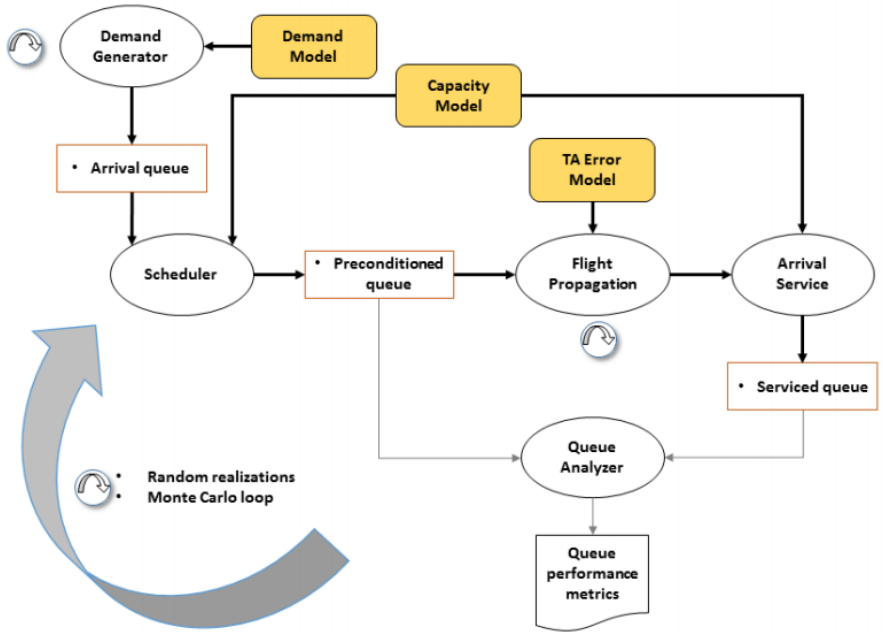

Figure 6 Qsim: Arrival Queue Simulator [42]

The Scheduler is the key component of the simulation system. It takes a set of estimated flight arrival times and assigns each flight an arrival slot with a corresponding CTA, following a FCFS sequence. The Capacity Model is used in this process to provide the AAR. When the CTA assigned is later than the ETA of the flight, an ETA delay occurs. This CTA assignment process in Qsim follows the FCFS rule, but in this dissertation, various other CTA assignment schemes are added.

In the real world, flights might encounter perturbations and hence not be able to meet their assigned CTAs exactly. The Flight Propagation module simulates this behavior after the flights cross the freeze horizon by adding random noise to the target CTAs to represent their actual times of arrival (ATAs). The Arrival Service models the arrival service based on given AARs. It puts one flight at a time in the queue in the order it

arrives and serves the next one in the queue when the server is freed up from the previous flight in the queue. When the time of service is later than the ATA, a service delay occurs. The Queue Analyzer collects the information from the previous steps and computes performance metrics such as average arrival delay (relative to ETAs) caused by assigning CTAs, average service delay, inter-arrival time variance, unused capacity, and queue lag time. The Qsim process loop includes all these previous components and was repeated for 5000 realizations to generate the performance metrics.

The simulation model used for this dissertation was developed using the original Qsim package as a foundation. The major simulation pipeline was ported from IGOR Pro into MATLAB, but variations of different components and more detailed data simulation procedures were applied (shown in Figure 7).

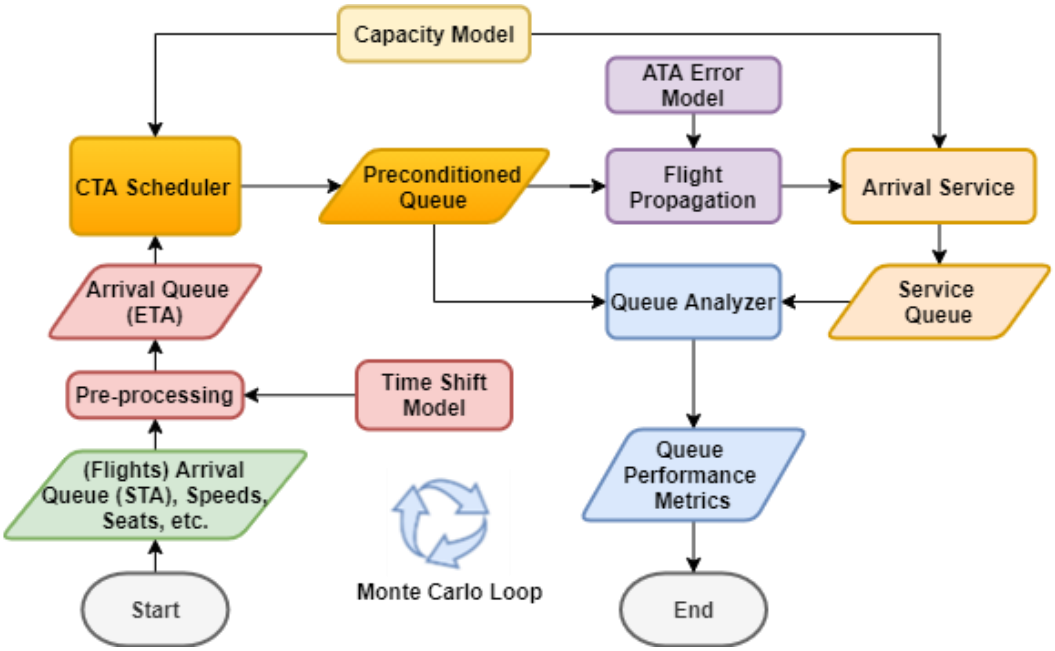

Figure 7 Simulation Structure

The major differences between our system and Qsim are summarized as follows:

1) Our simulation starts with a flight arrival queue containing a list of individual flights' STAs instead of aggregated flight counts data. Time shifts and noise were added to the STAs to mimic the ETAs following the procedures described in Section 4.2.2. This means our simulation is a more realistic representation of the real world, can be set up at any selected airport, and is not based on the performance on one particular historical day. Furthermore, our system is able to simulate a wider range of scenarios for the ETAs. For example, studies under different ETA distributions are conducted in this dissertation (further described in Section 5.5).

2) Our simulation incorporates more details to mimic more complicated and realistic scenarios. For example, speed limits and current speeds for each specific aircraft were

used when defining the available CTA slots, and wind speed was generated for each flight to mimic the real-world scenarios. Qsim does not differentiate these terms for different aircraft. Our simulation also allows different freeze horizon distances to explore extended metering and simulate a wider range of real-world scenarios (discussed in Section 5.7).

3) Instead of focusing on the delay of CTAs relative to ETAs, our system aimed at reducing the delay cost of CTAs relative to STAs, which is a performance metric more directly related to carriers' business objectives. We present the monetary values for these delays in the Queue Analyzer, which gives the CDM and TBFM participants a clearer view of how much they can save by adopting our schemes.

4) Our research proposes an "individual cost-based priority" scheme, allowing users to take into account the number of passengers, the number of connecting passengers, and other factors when generating different cost and defining priority preferences for different flights.

5) In addition to the FCFS scheme (baseline model), our system also provided a variation to FCFS that does not allow advancing, an optimization model, and two heuristics in the CTA assignment process (illustrated in Chapter 3). Even for the FCFS scheme, our simulations calculate different flights' different maximum advance times, instead of using one limit for all flights, as in Qsim. These schemes take account of information and preferences from different carriers and incorporate it into

the assignment to achieve a lower CTA delay cost, the cost associated with delays due to flow management measures en route and during arrival.

6) Our system adopted a rolling horizon approach which allows the CTA assignment scheme to operate dynamically throughout the day in small time intervals, rather than simulating a whole day's operations all together (described in Chapter 6). This makes the scheme more realistic and adaptable to any disruptions or changes in real time. The system also allows the usage of different values in AARs in the CTA scheduler and the arrival service. Additionally, fairness adjustment methods using real-time metric to achieve an overall fairer benefits distribution among different carriers are designed.

7) Our simulations explore different parameter and scenario settings and presented simulation results accordingly. This tells us the best parameters to use and the model performance in different scenarios.

4.2 Data Preparation and Simulation Procedures

The data inputs to our study are broken into a table of flights, a table of aircraft seats by aircraft type, a table of aircraft speeds by aircraft type, a table of wind speeds, a table of Airport Acceptance Rates (AARs), and other data. These are described in more detail in the following subsections.

4.2.1 Flights Data and Carrier Distribution

Atlanta Hartsfield-Jackson International Airport (ATL) was chosen as the study airport due to its high traffic volumes and extensive use of TBFM for approach traffic smoothing, but as stated above, the approach is generally applicable to any airport of interest.

A flight table of 1302 flights was retrieved from FlightAware for all flights having STAs on Monday, October 7, 2019 at ATL [43]. The chosen day was a typical traffic day at the airport with no severe weather conditions. The data fields used in this table include carrier, flight ID, aircraft type, scheduled arrival times (landing), and flight plan route distance. The STAs to the metering fix were approximated by using these scheduled arrival times. All timing data used in the project were converted to the number of seconds past midnight, Eastern Daylight Time.

A pie chart that shows the carrier distribution for the flight's dataset is given in Figure 8. Carriers that operate 5 or fewer flights are labeled as "Others" in this graph. We can see that the distribution of carriers is uneven. ATL is dominated by Delta Air Lines (DAL) in terms of the number of flights on the day. We thus designed our simulation to test if this uneven distribution brings unforeseen challenges to our model (more discussions in Section 5.6, Section 6.4, and Section 7.4).

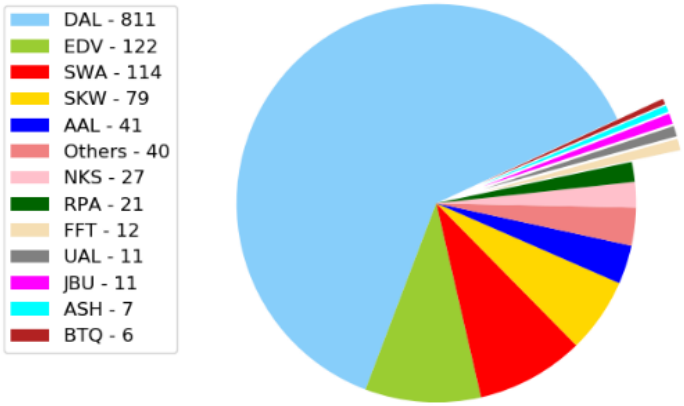

Figure 8 Number of Flights by Carrier Distribution

4.2.2 Flight ETA Generation

The primary inputs to our FCFS scheduler should be the flights' ETAs to the metering fix estimated at the time the flight crosses a reference boundary (e.g., the Freeze Horizon) from the airport. In this simulation, the FH was assumed to be 250 NM upstream of the metering fix.

Since these ETAs from the exact boundaries are not given, we simulate them by adding an exponential random variable to the flights' STAs. In this way, we reconstruct the state of the flow prior to applying the FCFS queuing rule to assign CTAs. The exponential distribution was chosen from analysis of the departure delay distribution at ATL airport on the same day [37]. The parameter used in this distribution can be adjusted for different delay scenarios.

There are other factors such as wind, pilot behavior, and ATC actions that can affect flights' ETAs, among which wind is the major factor. We thus refined this ETA

simulation model by adding a zero mean Gaussian component to account for wind deviations using the same method and parameters as described in [37, 44]. The previously described exponential distribution adds only a positive time shift, which makes sense for departure delays, while this Gaussian component can add both positive and negative noise to simulate the ETAs, which is an appropriate representation for wind noise, since the wind can blow in any direction. By changing the parameters of the Gaussian distribution, this model could be tailored to wind conditions that prevail at any particular airport.

These ETA simulating processes can be written as Eq. (13):

$$t_i^{\mathrm{ETA}} = t_i^{\mathrm{STA}} + t_i^1 + t_i^2 \tag{13}$$

where t_i^{ETA} is the ETA of flight i, t_i^{STA} is the STA of flight i, $t_i^1 \sim \exp(\lambda)$ represents the simulated arrival delay time shift for flight i, following an exponential distribution with rate parameter λ, and $t_i^2 \sim N(0, \sigma^2)$ represents the Gaussian noise that accounts for wind deviations [45], with parameters $\sigma = T_i/64$, and T_i is the time from departure to the FH for flight i, and is estimated as $max(t_i^{\mathrm{tt}} - 38\ \mathrm{min}, 0)$, with t_i^{tt} being the flight total travel time and 38 min being the nominal FH to the metering fix flying time.

This whole flight ETA generation process corresponds to the Time Shift Model, Pre-processing, and Arrival Queue (ETA) in Figure 7 in Section 4.1 (highlighted in red color). In this way, the incoming flight ETAs are constructed based on an empirically calibrated distribution that reflects various real-world conditions.

4.2.3 Flight Delay Cost Generation

The CTA assignment models need input data including flight delay cost and priority. Ideally, we would want carriers to provide this information following a Collaborative Decision-Making paradigm. In this dissertation, these values are simulated following methods described in this section and the following sections.

Our research simulates the aircraft passenger load factor and percentage of connecting passengers (conn_pax %) in order to obtain the number of passengers and number of connecting passengers in each aircraft.

Aircraft passenger load factors are generated following a beta (8, 2) distribution. This gives a distribution with a mean of 0.8 and a variance of 0.0145. The number of passengers is calculated by simply multiplying the load factor by the number of seats for each flight in the list. Since load factors are proportions, it is well-matched to the beta distribution, whose parameters allow for a wide variety of density function shapes within the domain [0, 1].

The percentages of connecting passengers are generated differently in different carriers using the parameters in Table 6. The means and variances of these distributions are also presented in Table 6. By following these distributions, we allow the “dominating” carriers to tend to have more connecting passengers than carriers operating only a few flights per day at the airport. The number of connecting passengers is calculated by multiplying the percentage of connecting passengers by the number of passengers for each flight in the list.

Table 6 Percentage of Connecting Passengers Distribution

Number of flights *n* within the carrier	conn_pax % Distribution	(mean, variance)
***n* ≥ 200**	beta (2,7)	(0.22, 0.02)
100 ≤ *n* < 200	beta (1.5,7)	(0.18, 0.02)
20 ≤ *n* < 100	beta (1,7)	(0.13, 0.01)
***n* < 20**	beta (0.5,7)	(0.07, 0.01)

After determining the number of passengers, the number of non-connecting passengers, and the number of connecting passengers, the delay cost for each flight can be calculated. The CTA delay in minutes of flight i is defined as the CTA delay relative to its STA and can be calculated using Eq.(9) in Section 3.2.2. The delay cost coefficient can be calculated using Eq.(8). In our simulation, the following values for these parameters are used: $c^{\text{pax}} = 0.4$ ($/pax/min); $c^{\text{connpax}} = 0.8$ ($/pax/min). These parameters are for simulation purposes and are drawn from previous studies [37, 46, 47]. They are also based on the understanding that delays to connecting passengers cost more than non-connecting passengers due to the risk of missing connecting flights. For simplicity, other costs c_i^o is simulated following a uniform distribution, $U(0, 4)$. The fuel cost is excluded in our simulations at this moment due to data access. For each flight, the CTA delay cost is then calculated by multiplying the CTA delay and the delay cost coefficient.

In real-world implementation, we might encounter challenges on obtaining airline cost information. But the proposed schemes can still be implemented by applying some alternative approaches for the cost coefficient part. For example, we could allow airlines to provide ranked lists to indicate the relative delay cost of their flights,

where a higher ranking indicates higher cost for a flight. We can then standardize these rankings within each carrier and use this as a swapping criterion instead of the real cost value, even among different carriers.

4.3.4 Flight Priority Generation

This section provides some simple approaches and rules for generating the flight priority labels. In reality, each carrier can provide their priority labels for their own flights as long as they follow these rules: the priority labels range from 1 (lowest priority) to 5 (highest priority); each priority group should contain (as close as possible) 20% of the carrier's flights at the airport; and the mean value of all the priority labels should be 3. These rules ensure our scheme does not allow for gaming in priority generation (e.g., carriers cannot just assign all flights to be the highest priority 5 because that will violate these rules).

To generate their priority lists, airlines can take into consideration a number of relevant factors, such as the number of connecting passengers within a time range, aircraft types and fuel consumption, number of seats and passengers, and deviations from their STAs. The information used by the airline to generate their priority list can remain private. The only required output from the airline is the priority list.

In this study, the priority labels are generated for flights based on their simulated individual cost using the following procedures: First, flights are sorted within each

carrier by cost from lowest to highest; second, we assign the top 20% on the flight list (take the floor if the number not an integer) to be priority 1, the following 20% to priority 2, …, the last 20% to priority 5. When the total number of flights cannot be divided by 5 (including smaller than 5), we add the number of flights in some priority groups in addition to this 20% portion:

Divide the total number of flights for the carrier by 5 and retrieve the remainder *r*.

- If *r* = 1, add 1 extra flight to the following priority group: [3]
- If *r* = 2, add 1 extra flight to each of the following priority group: [2, 4]
- If *r* = 3, add 1 extra flight to each of the following priority group: [2, 3, 4]
- If *r* = 4, add 1 extra flight to each of the following priority group: [1, 2, 4, 5]

The algorithm for this priority generation process is given in Table 7:

Table 7 Algorithm for Priority Label Generation

1. group flightsTable by carrier
2. for each carrier in the carrier list do
2.1 Flts = all flights belong to the carrier 2.2 nFlts = total number of flights in Flts 2.3 sort Flts by cost, ascending 2.4 r = nFlts mod 5 2.5 d = ⌊nFlts/5⌋ //floor
// define division points 2.6 div1 = d + 1*(r == 4) 2.7 div2 = div1 + d + 1*(r == 2||r == 3||r == 4) 2.8 div3 = div2 + d + 1*(r == 1||r == 3) 2.9 div4 = div3 + d + 1*(r == 2||r == 3||r == 4)
// generate priority labels 2.10 for flights from row 1 to div1 in Flts priority = 1 2.11 for flights from row div1+1 to div2 in Flts priority = 2 2.12 for flights from row div2+1 to div3 in Flts priority = 3 2.13 for flights from row div3+1 to div4 in Flts

```
    priority(j) = 4
  2.14 for flights from row div4+1 to nFlts in Flts
    priority = 5
3. combine all the carriers groups back together as flightsTable
4. sort flightsTable by ETA, ascending
```

4.4.5 Aircraft Number of Seats Table

A list of unique aircraft types was extracted from the flights table described previously. There are 37 unique aircraft types amongst all the flights in our input table. For modeling purposes, we assigned a single value for the typical number of seats for each aircraft type. This ignores some of the variations in aircraft cabin design, but such deviations are small and negligible for the purposes described here. These numbers are sufficient to represent large-scale differences between aircraft types. In a real-world application, airlines could provide their actual number of seats for aircraft. This information is important because it can serve as some representation of the "importance" of an aircraft in a modeling or practical scenario where occupancy information is not publicly available, or where other possible expressions of carrier preference are not feasible.

Most of the number of seats values used in this study come from an aircraft overview booklet by DVB Bank SE [48]. For those aircraft types not listed in this reference, additional information was found in the FAA Aircraft Registry, FlyRadius and Controller websites [49-51].

4.5.6 Speed Table

For each flight in the simulation, it is necessary to have three speed values at hand: the current speed that a particular simulated flight will be assumed to follow, and the minimum and maximum speeds that it could be assigned. These speed extrema allow the model to determine the range of CTA slots that are feasible for an aircraft, in the event that swapping flights between slots can improve the overall objective function.

a) Aircraft Speed Extrema

The values for the minimum and maximum possible speed for each aircraft type were determined empirically based on a margin of speeds that is operationally reasonable. For aircraft that can fly this high, a default of 35,000 ft. was chosen as the cruising altitude, presuming this is the nominal altitude at the freeze horizon. Historical average altitude was used for other aircraft. Although TBFM applies to a phase of flight where altitudes and speeds are expected to change as part of the descent, a constant speed assumption is sufficient to distinguish the aircraft types. In a real environment, actual speeds would be known, and it is possible that carriers could indicate desired speed extrema based on fuel consumption or other economic factors, rather than more liberal controller limits or aircraft operational envelopes.

For a limited number of aircraft that could not be categorized this way, the values recommended for similar aircraft and ICAO's DOC 8643 Designators were used [52].

b) Aircraft Current Speed

The current speed assigned to each flight is sampled from a distribution of historical flight speeds at the appropriate altitude for the corresponding aircraft type. A large sample of historical flight plans (submitted in August 2017) was analyzed to determine average speeds filed for cruise. A sample of the analyzed statistics is shown in Table 8.

Table 8 Flight Plan Speeds by Aircraft Type

actype	hnpt	havg (ft)	…	avgtas (nm/hr)
A319	45,497	34,042		451
A320	71,861	33,894		456
A321	38,812	32,846		459
…	…	…	…	…

where *actype* = aircraft type (ICAO designator as filed); *hnpt* = number of flight plans analyzed; *havg* = average cruise altitude filed (in feet); *avgtas* = average true airspeed (TAS) at altitude *havg*.

Note that some aircraft types cannot fly at 35,000 feet (B190, BE58, BE9L, C208, and PC12). For these aircraft types, the average speed at 35,000 feet is not available and should not be used. Instead, the average TAS at their historical average altitude are used.

The speed table was joined to the flight table to indicate for each flight its current, minimum, and maximum speeds. For the purposes of simulation, zero-mean Gaussian noise with standard deviation of 3.33 knots was added to the existing current speed distribution to approximate the flight's true current speed at the freeze horizon. This

step allows different flights on the list to have slightly different current speeds even if they belong to the same aircraft type. With this choice of standard deviation, approximately 99.7% of the current speed values will be within 10 knots of the historical flight plan *avgtas* value for each aircraft type. A value check is conducted to make sure each aircraft's new current speed is within the speed limit range. If it violates the maximum or the minimum speed check, it is clamped to the closest boundary value.

c) Wind Speed

To simulate the real-world environment, it is important to take wind into consideration. The speeds indicated in the previous steps are all TAS. In order to relate the results to the ground speed, the wind speed then needs to be incorporated.

To characterize wind speeds encountered by flights in cruise (near the freeze horizon) going into ATL, a large sample of flight arrivals and the along-track wind for each trajectory at ATL were analyzed using publicly available data [53]. The set of trajectories is shown in Figure 9. Figure 10 shows the histogram of the along-track wind speed for the altitude bin [30,000 - 40,000] feet. Since Figure 9 shows clustering of trajectories, it is not surprising that there is also some bunching of speeds in Figure 10.

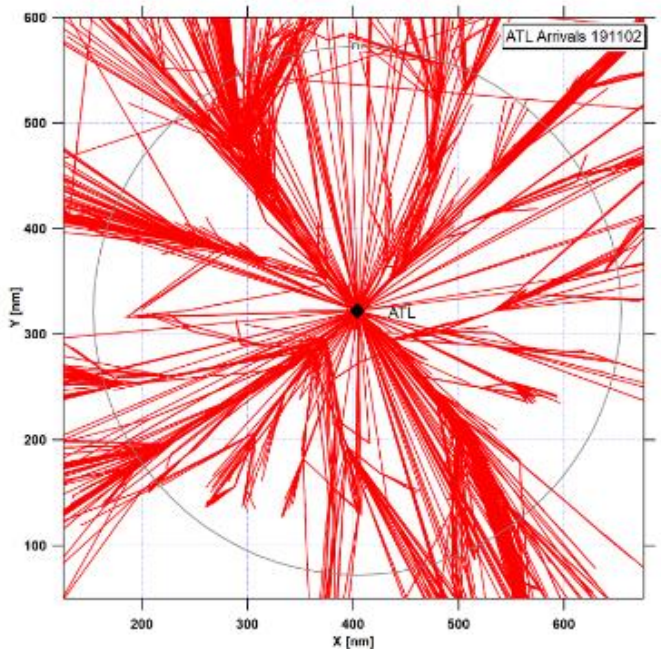

Figure 9 Trajectory Footprint for Selected ATL Arrivals (11/02/2019) [53]

In an effort to keep the fidelity of the simulation at a manageable level, it was decided not to try to correlate trajectories to speed distributions, particularly because while these things were true for this particular historical day, something else would be true on other days. For example, there is seasonal variation, with lower wind speed values expected in warmer air. The simplest way to simulate the wind distribution was to use a uniform distribution. We chose a distribution of $U(-70, 70)$ to use in simulations after observing the empirical data. Wind speed was generated for each flight following this distribution and added to the three TAS columns (maximum speeds, minimum speeds, and current speeds) in the flights table to represent the ground speed.

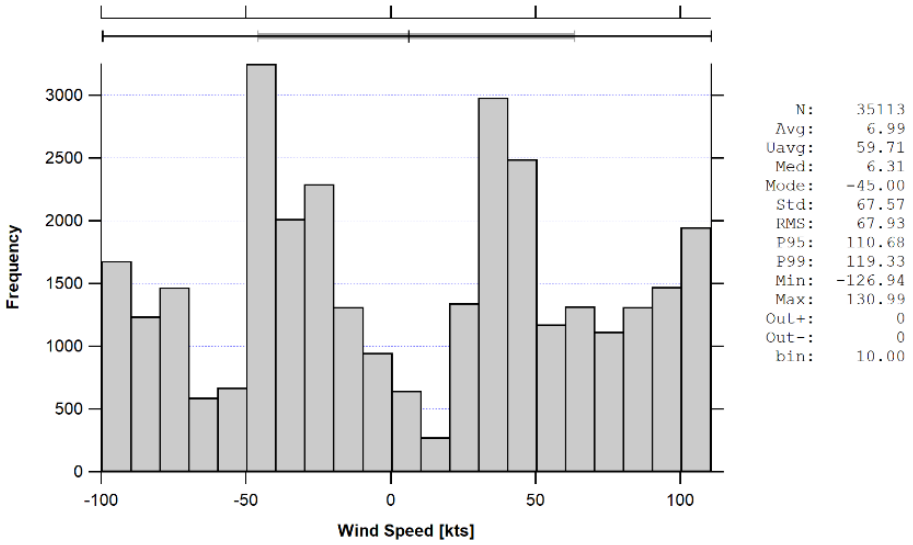

Figure 10 Along-track Wind Speed Distribution for Altitude Bin [30000,40000] Feet [53]

In future implementations, this wind generation model can be replaced with a more comprehensive model that captures the dependency of nearby aircraft traveling in the same direction, if applicable.

4.5.7 Airport Arrival Rate

AAR is a dynamic parameter specifying the number of aircraft that an airport, in conjunction with terminal airspace, can accept under specific conditions throughout any consecutive 60-minute period [54]. The AAR for a given airport varies in different weather conditions and runway configurations and is used in the Capacity Model in our simulation (highlighted in light yellow in Figure 7). Historical data from Aviation System Performance Metrics (ASPM) show that the average AAR at ATL on 10/07/2019 is 106 flights/hour [55]. To make our simulation more applicable to any other day situations, simulations on a range of different AAR values were

performed. A full table for the ATL AAR was obtained from the FAA website [56] (a sample is shown in Table 9). Among all weather and runway configuration combinations, the maximum AAR of all conditions is 132 and the minimum AAR of all conditions is 18. Because in real-world operations, highly imbalanced demand and AAR will initiate Traffic Management Initiatives such as GS and GDP, the AAR and demand in TBFM should always remain comparable. In this study, we thus omit extreme conditions and implement our models under an AAR range of [80, 140] flights/hour with a step size of 10 flights/hour to simulate a full range of operational conditions at ATL.

Historical AAR data at ATL from ASPM were also retrieved [55]. The hourly distribution of AAR at ATL during 2015 - 2019 is shown in Figure 11. The minimum value of these historical AAR values is 16, and the maximum value is 132. Our simulated AAR range [80, 140] covers 99.64% of the historical scenarios.

Table 9 ATL AAR Information

Runway configuration		Meteorological Condition			
arrival	departure	VMC (3600/7)	LOW VMC	IMC	LOW IMC
26R|27L|28	26L|27R	132	124	110	98
26R|27R|28	26L|27L	122	116	104	96
…	…	…	…	…	…
09L		50	45	40	36
Data source: https://www.fly.faa.gov/Information/east/ztl/atl/atl_aar.htm					

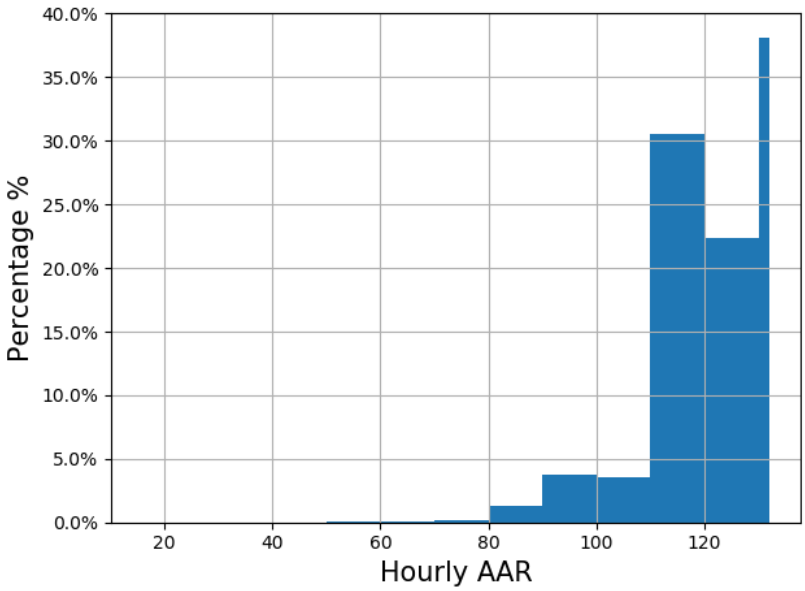

Figure 11 Hourly AAR Distribution for ATL during 2015-2019

There were no extreme weather conditions in Atlanta on 10/07/2019. Hence, our historical traffic data from that day represent a high traffic scenario.

4.5.8 Flight ATA Generation

In real-world conditions, an aircraft has meet time errors, and there are various perturbations that a flight might encounter from the time a CTA has been assigned to the time when it reaches the metering fix. When a flight arrives at the metering fix, its ATA could be ahead of its CTA, behind its CTA, or exactly the same as its CTA. When a flight arrives and the landing services are still occupied from preceding flights, the final approach and tower controllers may need to send the flight into a delay absorption deviation or holding. This is when a service delay occurs. To model these time deviations, Gaussian random variables are added to the assigned CTAs to represent flights' ATAs. This model is referred to as the "ATA Error Model" and the

process is referred to as “Flight Propagation” in Figure 7 (highlighted in purple) and can be calculated following Eq.(14).

$$t_i^{\mathrm{ATA}} = t_i^{\mathrm{CTA}} + t_i^{\mathrm{noise}} \tag{14}$$

where t_i^{CTA} is the CTA of flight i, t_i^{ATA} is the ATA of flight i, $t_i^{\mathrm{noise}} \sim N(\mu, \sigma^2)$ represents the simulated noise for flight i, following a Gaussian distribution. The parameters in this distribution can be varied to simulate different noise scenarios.

As part of the performance metrics, the service delay and service delay cost are also calculated in our simulations. For simplicity, the same cost coefficient as in the CTA delay cost is used for the service delay cost.

4.3 Simulation Environment

All the CTA assignment schemes were coded in MATLAB and the optimization model required MATLAB Optimization Toolbox. The Monte Carlo simulations were conducted in MATLAB 2020b on the Deepthought2 High-Performance Computing clusters at the University of Maryland, College Park using 12-process parallel computing. The random numbers were controlled in a way such that each Monte Carlo realization uses its own particular set of random numbers, regardless of which worker runs that realization or what sequence the realizations run in. The single-realization simulations were conducted on a Lenovo laptop with a processor of Intel Core i7-8650U CPU using serial computing.

4.4 Chapter Conclusion

This chapter introduced the setup for our simulation system. The simulation system structure was first explained, followed by detailed data preparation and simulation procedures in each step. Following these instructions, the data from any day or any airport can be processed to be used in our system. The simulation environment was specified in the end. The corresponding simulations were then performed, and the results were documented in Chapter 5.

Chapter 5: Static Case Studies

This chapter presents static simulation results under several different scenarios and parameter settings.

5.1 Simulations Comparing the FCFS with and without Advancing

As introduced in Section 3.1, two FCFS schemes were proposed: FCFS, which allows advancing flights, and FCFS_w/o_ADV, which prohibits advancing flights. In this section, simulations are performed for these two FCFS schemes for 7 AAR values in [80, 90, 100, 110, 120, 130, 140]. The ATA error model used is Gaussian noise among all flights with mean = 0 s and standard deviation = 15 s. For each setting, 500 Monte Carlo realizations are conducted where slightly different flights tables are generated based on noise distributions to use as inputs. The average values of performance metrics from these 500 realizations are calculated and shown in Table 10. The total CTA delay cost represents the total CTA delay cost for all flights in the 24-hour period at ATL.

Table 10 Total CTA Delay Cost vs. AAR for Two Different FCFS Schemes ($)

AAR	80	90	100	110	120	130	140
Demand/Capacity	0.98	0.87	0.78	0.71	0.65	0.6	0.56
FCFS_w/o_ADV	961,565	360,939	176,320	123,635	94,585	77,580	65,971
FCFS	947,404	345,865	165,240	113,440	84,884	68,655	57,645
Reduction	14,161	15,074	11,080	10,195	9,701	8,925	8,326

Reduction %	1.47%	4.18%	6.28%	8.25%	10.26%	11.50%	12.62%

It can be seen that allowing advancing reduced the total CTA delay cost (shown in the differences between Row 3 and Row 4 in Table 10). This is as expected. Advancing flights in FCFS allows a flight to be assigned a CTA slot earlier than its ETA as long as it is within the aircraft's feasibility range, and thus reduces the total CTA delay and the corresponding cost. Under each scheme, as the AAR value increases, the capacity of the airport increases and the total CTA delay cost decreases, as expected. For future case studies, unless stated otherwise, we use "FCFS" to refer to the FCFS scheme with advancing and it is selected as the baseline model.

5.2 Simulations Exploring Different Neighbor Searching Sizes in 2OptSwap

The performance and runtime of our heuristic approach, 2OptSwap, depends on the parameter settings for the neighbor searching size (NSS). That is, for each flight, how many succeeding flights do we explore and try to swap the CTA with. In this section, simulation experiments are conducted to explore a set of different values for the neighbor searching size.

The priority penalty in the objective function for the IP optimization model and the carrier preference check for the 2OptSwap were excluded in this set of simulation experiments such that the results are comparable. A value of 10 min (between CTA in

FCFS and CTA in heuristic or IP solution) was selected as the max delay time for all flights.

The performance changes in terms of CTA delay cost and runtime in the 2OptSwap scheme are explored, for 7 AAR values in [80, 90, 100, 110, 120, 130, 140] flights/hour and 13 different NSS values in [5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, whole day flights]. The whole day flights option means we explore swapping each flight with all the succeeding flights until the end of the day. The ATA error model used is Gaussian noise among all flights with mean = 0 s and standard deviation = 15 s. For each setting, 500 Monte Carlo realizations were conducted where slightly different flights tables were generated to use as inputs. The mean values of performance metrics from these 500 realizations are calculated and used in the analysis. We also execute the IP optimization model to track the optimal solution for each AAR value and compare it with the trend from 2OptSwap.

First of all, a set of normality tests were conducted to check if our Monte Carlo simulations were performing as expected. We assumed the data from different Monte Carlo realizations should be normally distributed. As an example, we present the results for 2OptSwap under the setting of AAR = 100 aircraft/hour, ATA error model with noise $\sim N(0, 10)$, and NSS = 20 flights below. We first observed the shape of the empirical data distribution using graphical methods (Figure 12 and Figure 13), then performed the Kolmogorov-Smirnov test to validate our assumption. The Kolmogorov-Smirnov test (K-S test) computes the distance between the empirical data and a hypothetical known distribution and decides (in a rough sense) if the data

conform to the hypothetical distribution [57-59]. In Figure 12, the blue rectangles are the histogram from the empirical data from our Monte Carlo simulations (500 realizations). The red line is the density function of a normal distribution whose moments match our sample mean and sample variance. In Figure 13, the blue line is the cumulative distribution function (CDF) of our empirical data, and the red line is the normal distribution CDF of the given sample mean and sample variance. The K-S test statistic is the maximum absolute vertical difference between these two CDFs. The p-value of this K-S test is 0.42 (larger than 0.05); thus, we cannot reject the hypothesis that our empirical data follow such a normal distribution, and our Monte Carlo simulations are performing as expected.

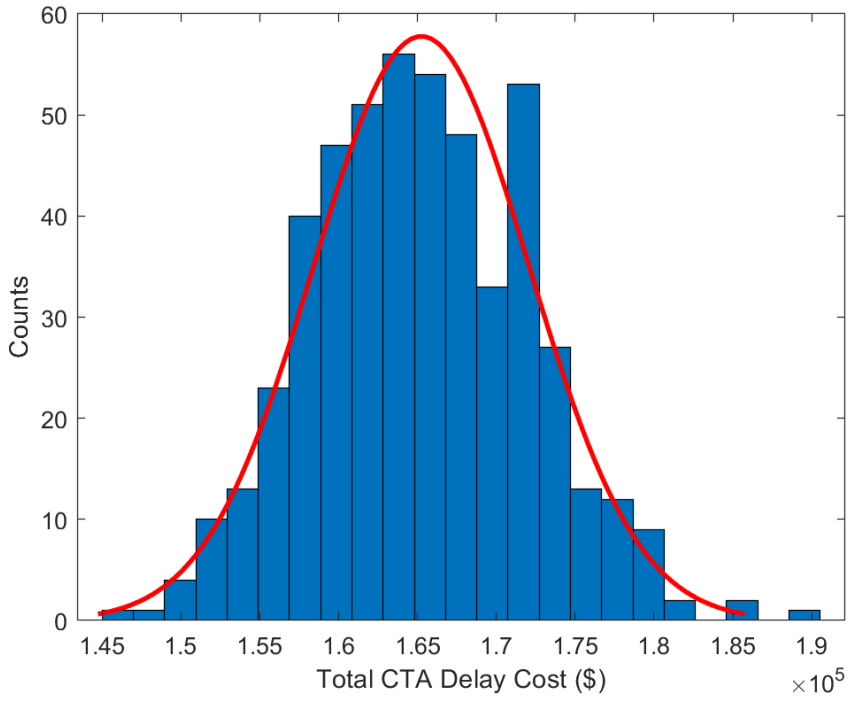

Figure 12 The Total CTA Delay Cost Distribution ($) over 500 Realizations

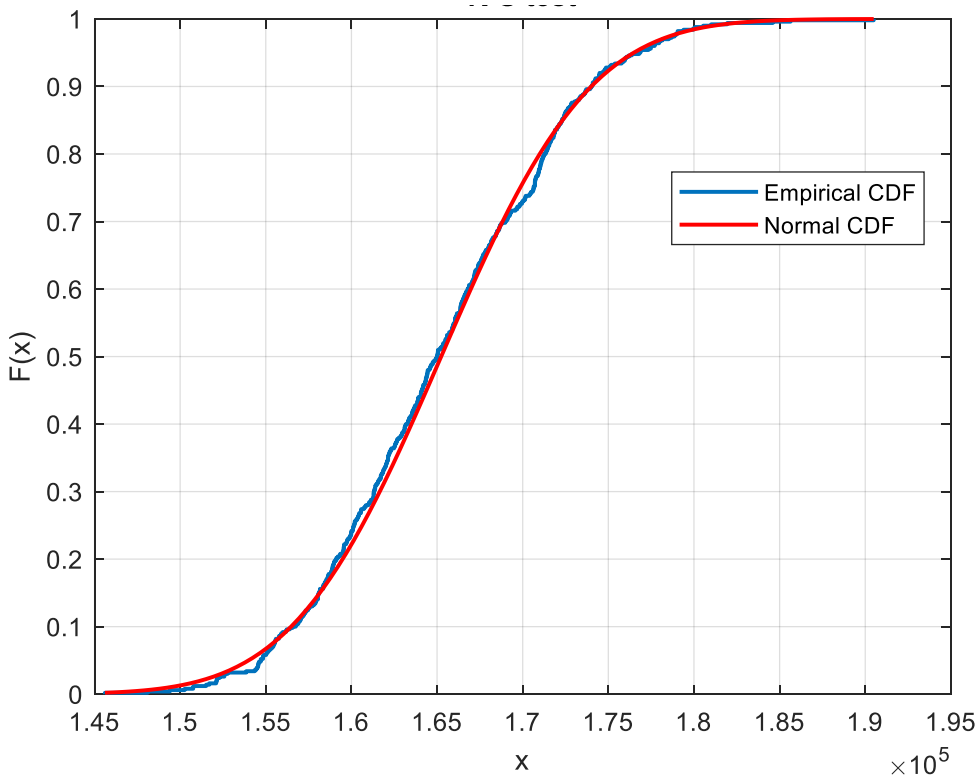

Figure 13 The CDF for the Empirical Data and the Normal Distribution

Secondly, the total CTA delay cost for different parameter settings is analyzed in Figure 14 and Table 11. It can be seen that as the neighbor searching size increases, the total CTA delay cost decreases and gradually gets close to the optimal solution in the IP model. The rate of decrease is high when the neighbor searching size is small (e.g., from 5 to 10), but it slows down as the searching size increases. The total CTA delay cost curve stays almost flat after the neighbor searching size increases to 15 or more in Figure 14. This is as expected because any swapping at this point will tend to violate the maximum delay and maximum advance time limit when the two flights are far away from each other in the ETA sequence. After a certain value, increasing

the neighbor searching size in the algorithm will just waste time exploring additional flight pairs while making no more swaps due to time feasibility.

In Table 11, we also included the setting that allows the algorithm to search the whole day of flights, and the results from the IP optimization model. Results show that even if searching all the succeeding flights until the end of the day, the results will only improve within 0.6%, compared to the results when the neighbor searching size is 20, for all the AAR values.

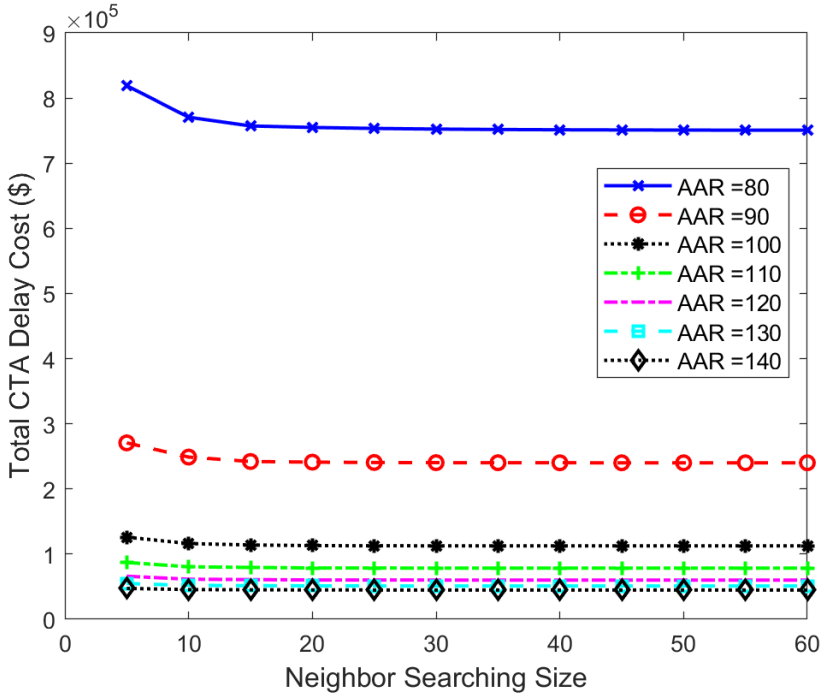

Figure 14 Total CTA Delay Cost in 2OptSwap vs. Neighbor Searching Size

Table 11 Total CTA Delay Cost in 2OptSwap vs. Neighbor Searching Size

	NSS = 10	NSS = 20	NSS = 30	NSS = 40	NSS = 50	NSS = 60	NSS = full table	IP
AAR = 80	770,581	754,907	752,407	751,334	750,853	750,664	750,627	742,074
AAR = 90	248,941	241,148	240,387	240,202	240,150	240,149	240,149	232,940
AAR = 100	116,185	112,996	112,708	112,634	112,632	112,632	112,632	107,682

AAR = 110	80,499	78,720	78,554	78,507	78,507	78,507	78,507	74,135
AAR = 120	61,452	60,312	60,188	60,157	60,157	60,157	60,157	55,902
AAR = 130	52,027	51,262	51,100	51,083	51,083	51,083	51,083	46,825
AAR = 140	45,565	45,011	44,886	44,879	44,879	44,879	44,879	40,573

Finally, the runtime for these different settings were compared. Figure 15 and Table 12 show that as the neighbor searching size increases, the runtime for 2OptSwap increases. This is as expected because each additional flights pair exploration costs time. Table 12 includes the runtime if searching the whole day flight table and it can be seen that it is computationally expensive to do so. To keep our heuristic approach efficient, we should choose a neighbor searching size relatively small such that it does not waste too much unnecessary computational time.

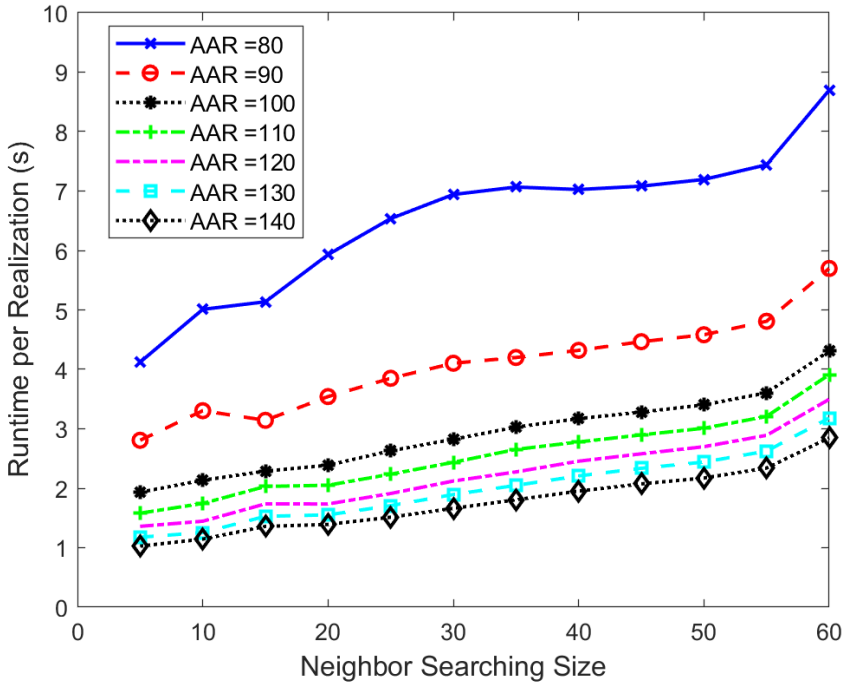

Figure 15 Runtime per Realization in 2OptSwap vs. Neighbor Searching Size

Table 12 Runtime per Realization in 2OptSwap vs. Neighbor Searching Size

Runtime (s)	NSS = 10	NSS = 20	NSS = 30	NSS = 40	NSS = 50	NSS = 60	NSS = full table	IP
AAR = 80	5.01	5.93	6.94	7.03	7.19	8.70	31.15	39.81
AAR = 90	3.30	3.54	4.10	4.32	4.58	5.70	25.61	40.17
AAR = 100	2.13	2.39	2.82	3.17	3.40	4.31	23.44	41.82
AAR = 110	1.74	2.05	2.44	2.78	3.01	3.91	22.69	44.83
AAR = 120	1.45	1.74	2.12	2.46	2.70	3.50	21.57	48.12
AAR = 130	1.25	1.55	1.89	2.21	2.44	3.18	20.56	51.13
AAR = 140	1.14	1.39	1.66	1.95	2.17	2.85	19.12	54.29

In summary, the neighbor searching size should be set at a reasonable value so that the heuristic can achieve relatively good solutions while not costing too much time computationally. Considering the above results and discussions, 20 is selected as the neighbor searching size for our 2OptSwap algorithm to be used in further case studies.

5.3 Simulations Comparing the Four CTA Assignment Schemes

In this section, the performance of the four CTA assignment schemes is compared: FCFS (baseline model), 2OptSwap, 2OptSwap_SA, and the IP optimization model. The carrier preference check in 2OptSwap and the priority penalty part in the objective function in the IP optimization model are removed, to make the results comparable. The weight for the ETA delay cost in the objective function, w^{ETA}, is set at 0.005 after testing a set of values. For parameters in 2OptSwap_SA, the initial temperature T is set at 1, the frozen temperature is set at 0.02, and the cooling ratio r

is set at 0.8. These values are selected after testing a set of values, considering the combination of runtime and performance.

Simulations for the four CTA assignment schemes are performed with 7 AAR values in [80, 90, 100, 110, 120, 130, 140] flights/hour. For each setting, 500 Monte Carlo realizations are conducted where slightly different flights tables are generated to use as inputs. The total CTA delay cost and the runtime are tracked for analysis. The mean value and the standard deviation of the metrics are calculated over the 500 realizations for each parameter setting.

Figure 16 shows the total CTA delay cost for different CTA assignment schemes. The mean value is used, and the standard deviation is plotted as error bars at each AAR value. It can be seen that the FCFS scheme generates the highest total CTA delay cost over all the models and the IP optimization scheme generates the lowest total CTA delay cost. The 2OptSwap and 2OptSwap_SA schemes generate almost the same total CTA delay cost, at a slightly higher value than the IP optimization scheme. For all the four schemes, as the AAR value increases, the total CTA delay cost decreases. This is

as expected because when the demand (incoming flights) stays the same, the higher capacity (represented by AAR) is, the lower delay and delay cost there will be.

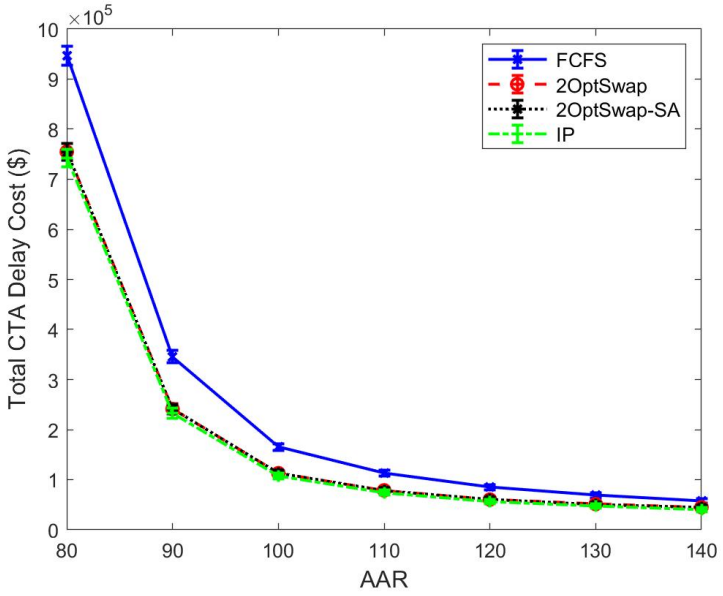

Figure 16 Total CTA Delay Cost for Different Schemes

The total CTA delay cost reduction values and percentages compared to FCFS for the three other schemes are presented in Figure 17 and Figure 19. In Figure 17, the mean value is used, and the standard deviation is plotted as error bars at each AAR value. It can be seen that the IP optimization model always provides the highest reduction and the 2OptSwap and 2OptSwap_SA provide almost the same reduction from the baseline model, at a lower value. This is as expected because the heuristics could face the problem of being trapped in local optima. For the same minimization problem, the optimal solution is always a lower bound to the heuristics. For the cost reduction

compared to FCFS here, the IP optimization model then provides the upper bound in terms of relative reduction value.

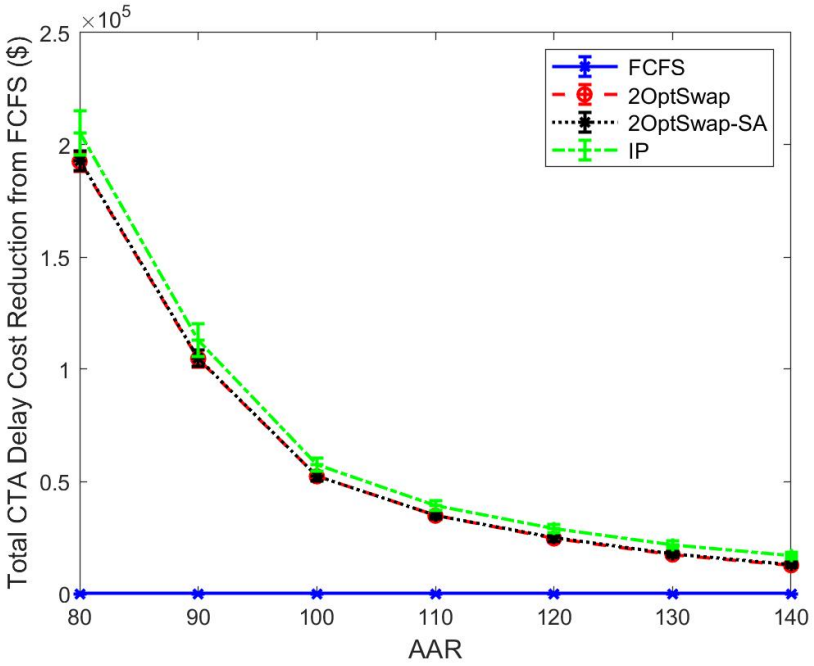

Figure 17 Total CTA Delay Cost Reduction Value Compared to FCFS

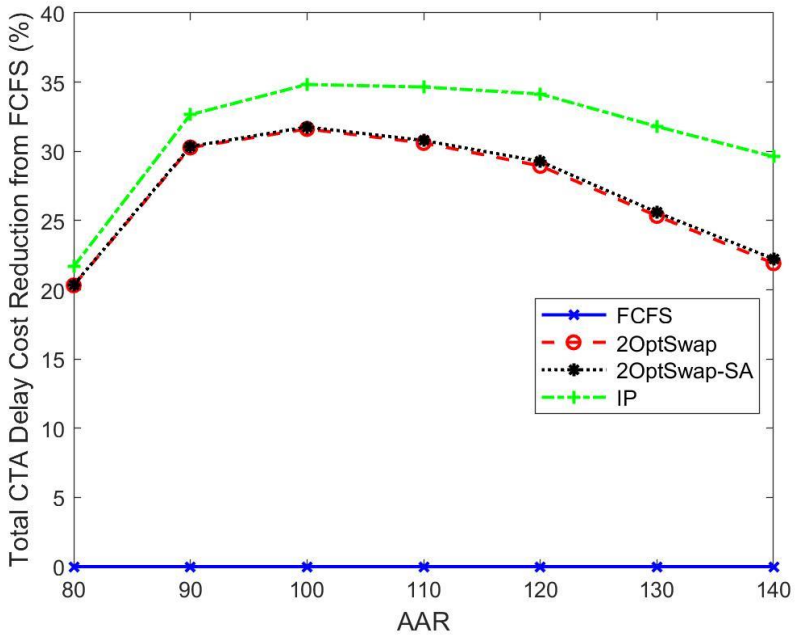

Figure 18 Total CTA Delay Cost Reduction Percentage Compared to FCFS

To better compare our heuristics approach with the optimization model in solution quality, the total CTA delay cost reduction from the heuristics to the optimization model is presented in Table 13. The two heuristic approaches, 2OptSwap and 2OptSwap_SA, both revealed promising results (the optimal solution is only 1% to 10% away from these heuristics solutions). The 2OptSwap reaches a 5.93% gap on average, compared to the optimal solution. And the 2OptSwap reaches a 5.69% gap on average, compared to the optimal solution.

Table 13 Total CTA Delay Cost Reduction from Heuristics to Optimization

		80	90	100	110	120	130	140
Reduction (2OptSwap - IP)	$	12,833	8,209	5,314	4,585	4,410	4,437	4,438
	%	1.7%	3.4%	4.7%	5.8%	7.3%	8.7%	9.9%
Reduction (2OptSwap_SA - IP)	$	12,544	7,953	5,078	4,381	4,136	4,258	4,259
	%	1.7%	3.3%	4.5%	5.6%	6.9%	8.3%	9.5%

The runtime for CTA assignment and landing service in each Monte Carlo realization and the mean value for all the realizations are shown in Table 14 and Figure 19. As the AAR value increases, the runtime in the IP optimization model increases while the runtime in the heuristics decreases. Between the two heuristics, the 2OptSwap_SA is much slower than the 2OptSwap, but we need to keep in mind that the runtime for 2OptSwap_SA values is sensitive to the parameter settings in the simulated annealing algorithm, such as the stopping criteria.

Table 14 Runtime per Realization (s) for Different Schemes and AARs

AAR	80	90	100	110	120	130	140
FCFS	0.04	0.04	0.04	0.04	0.04	0.04	0.04
2OptSwap	5.62	3.48	2.38	2.02	1.71	1.54	1.39
2OptSwap_SA	14.31	11.09	8.16	7.04	6.28	5.80	5.38
IP	39.86	40.36	41.93	44.90	48.10	51.16	54.27

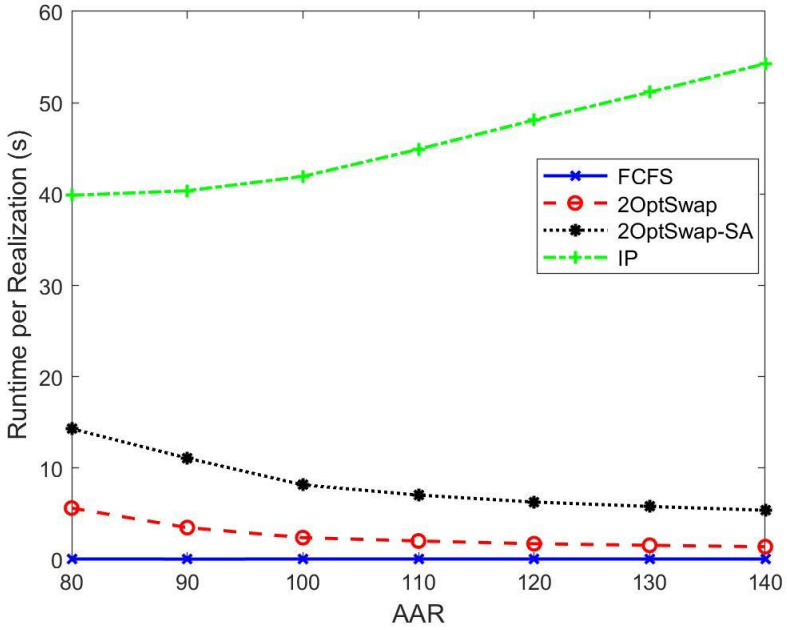

Figure 19 Runtime per Realization (s) for Different Schemes

Table 15 shows the ratios of the heuristics runtime to the IP optimization runtime. Our 2OptSwap approach is 7 to 40 times faster than IP optimization in runtime. This makes the 2OptSwap approach advantageous when real-time solutions are needed. The 2OptSwap_SA approach is 2 to 10 times faster than the IP optimization in runtime. Taking both the solution quality and the runtime comparison of these two heuristics into consideration, 2OptSwap is much faster than 2OptSwap_SA yet still provides almost the same results in total CTA delay cost. We thus keep 2OptSwap as the selected heuristic for further simulations. However, future studies could consider testing a more comprehensive range of different parameter settings in 2OptSwap_SA.

Table 15 Heuristics Runtime per Realization (Ratio to IP Runtime)

AAR	80	90	100	110	120	130	140
2OptSwap	7.10	11.58	17.60	22.27	28.05	33.15	39.11
2OptSwap_SA	2.79	3.64	5.14	6.38	7.66	8.83	10.09

5.4 Simulations Comparing the FCFS and 2OptSwap Schemes

In this set of simulation experiments, the carrier preference check is added back into the 2OptSwap scheme and the performance for the 2OptSwap approach under different parameter settings are explored.

A single-realization simulation was conducted to illustrate the simulation process and metrics. The parameters used are as follows: AAR = 84 flights/hour; ATA error model: Gaussian noise among all flights with mean = 0 s and standard deviation = 15 s. Simulations for the two different CTA assignment schemes are conducted separately.

For the 2OptSwap scheme, the number of swaps completed, the total CTA delay, and the total CTA delay cost are tracked in each iteration and the results are shown in Figure 20. It can be seen that the number of swaps is initially high at the first iteration but gradually reduced to zero and that is when the algorithm stops. The total CTA delay was reduced with each swapping iteration until there were no more swaps completed. A comparison of the results between the two schemes is given in Table 16.

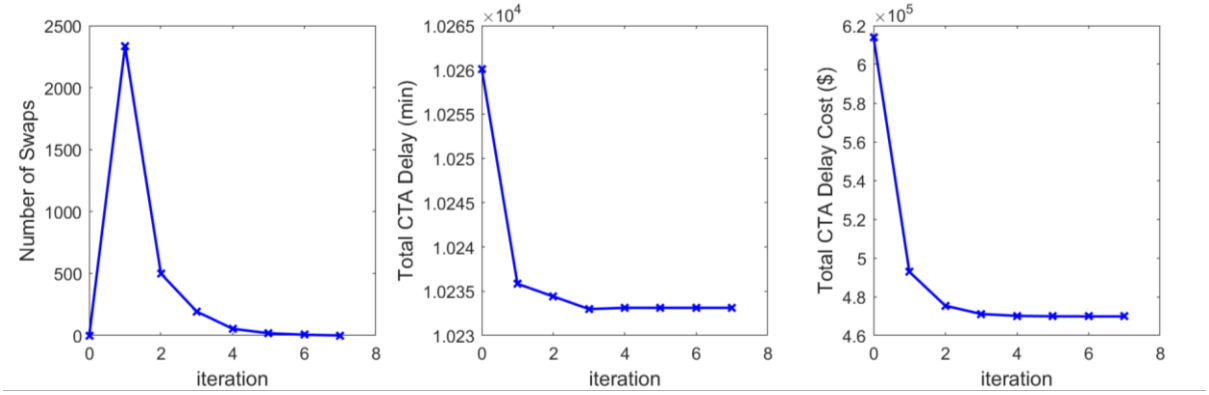

Figure 20 Metrics vs. Iteration in 2OptSwap

In addition to the total CTA delay cost, the metrics also tracked the total service delay cost for the two schemes. The service delay occurs because of the ATA error model used in generating ATAs from the CTAs. When a flight arrives at the airport and the landing services are still occupied from preceding flights, it has to wait to land and a service delay occurs. The same cost coefficient as in the CTA delay cost is used for the service delay cost.

The results for this single-realization simulation are shown in Table 16, Figure 21 and Figure 22. In Table 16, it can be observed that there is a reduction of 144 thousand dollars and 23% for the total CTA delay cost by conducting 2OptSwap, compared to the FCFS only scheme. The total service delays between the two schemes vary only by 472 dollars and 1.4%, which is as expected since the only source of this delay comes from the random ATA error model noise generated after CTA assignment and should not be affected by swapped CTAs among flights.

Table 16 Single-Realization Simulation Results: Comparing the Two CTA Assignment Schemes

	FCFS	2OptSwap	Reduction	Reduction %

Total service delay cost ($):	33,552	33,080	472	1.4%
Total CTA delay cost ($):	613,952	469,847	144,105	23.5%

Figure 21 and Figure 22 show the selected metrics over the 24-hour simulation time. The x-axis represents the time of the day. In the top subplots, the y-axis is the flights count per 15 minutes, and each light pink bar represents the demand of incoming flights for a 15-minute interval. We can see how the demand changes throughout the day and how the AAR serves as an upper bound (21 flights/15-minute in this case) for the CTA assignment/schedule rate (green line in the top subplots). In the bottom subplots, we can see that during some time periods, especially during peak hours, the CTA schedule delay (green line) is different between Figure 21 and Figure 22, but the service delay (brown line) is almost identical between the two plots. These observations are consistent with the values in Table 16. Note that these results are from one realization and are for illustrative purposes only.

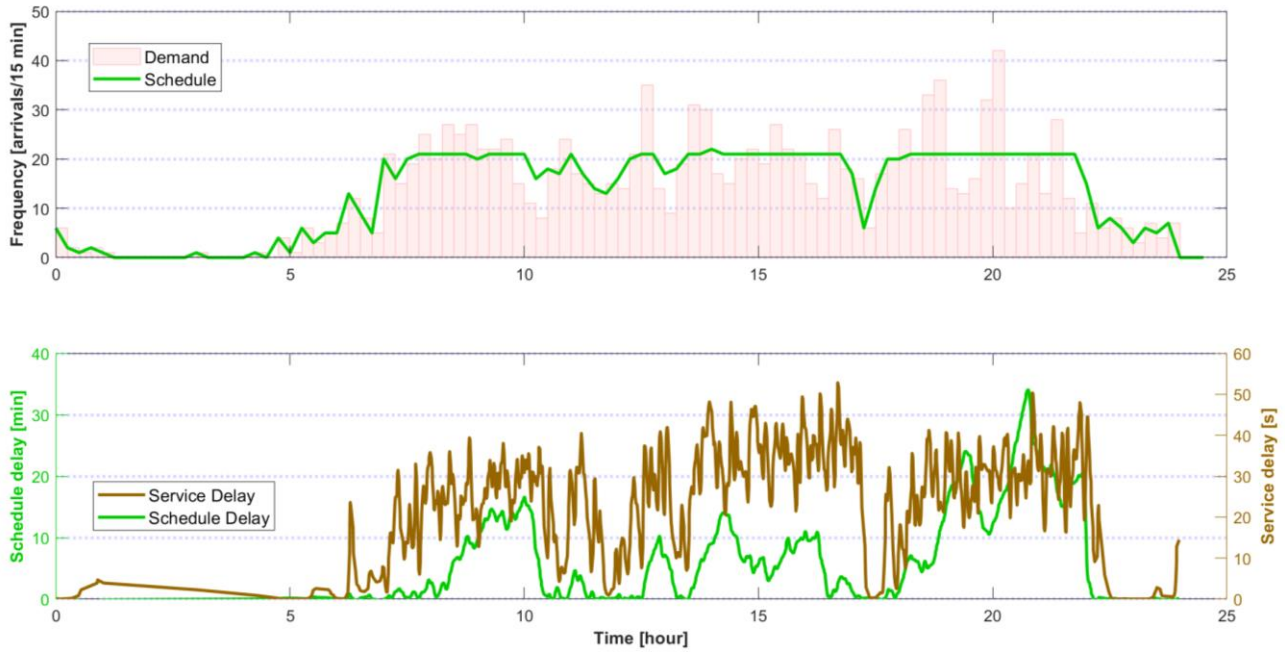

Figure 21 Single-Realization Simulation Results - FCFS

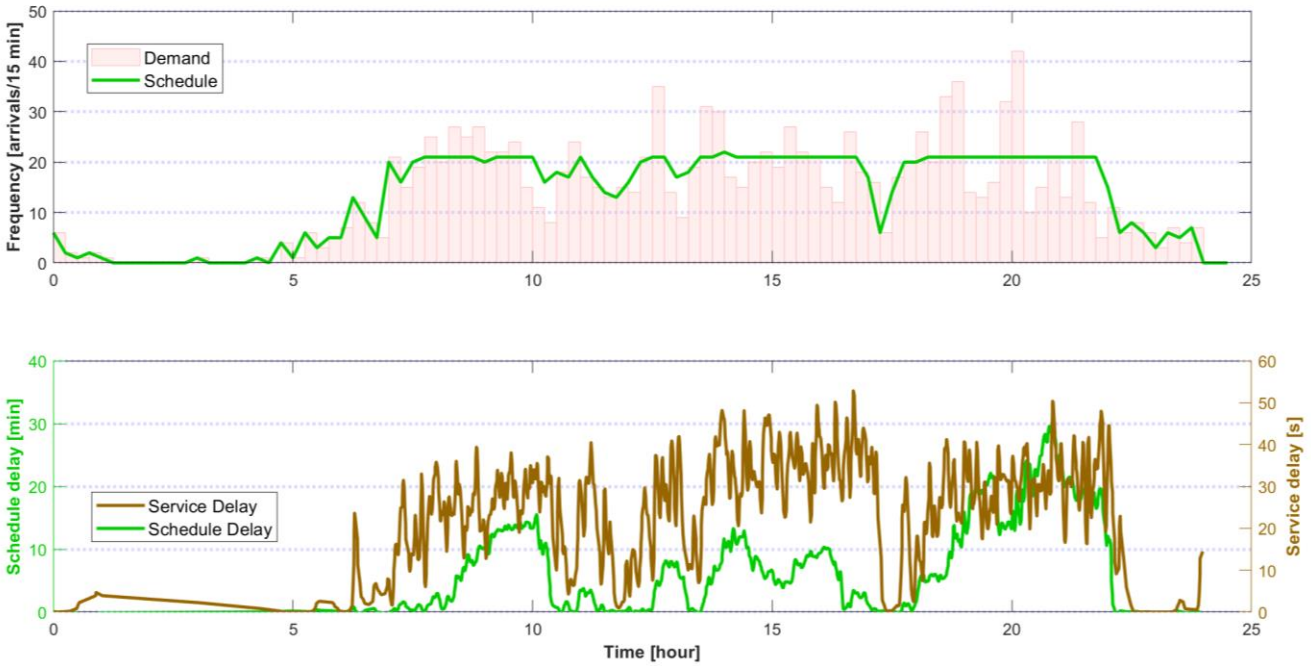

Figure 22 Single-Realization Simulation Results - 2OptSwap

To further test these two CTA assignment schemes, a second set of simulations was conducted by repeating a Monte Carlo loop (shown in Figure 23) for 500 realizations, for each of 7 different AAR values (80 to 140 flights/hour), 5 zero-mean Gaussian noise distributions with standard deviations of 0, 10, 20, 30, and 40 seconds as the 5 ATA error models. All aircraft in each model were treated to follow the same distribution. In real life, the exact form of the noise should depend on the phase of the flight and on different aircraft's navigation capabilities. The total CTA delay cost and the total service delay cost are tracked.

Similar to the single-realization simulation above, it is not surprising that we do not observe a significant difference for the total service delay cost between the two CTA assignment schemes, given that they use the same AAR and the same standard deviation of the ATA error model (shown in Figure 23). In fact, the difference in percentage of the total service delay between the two schemes stayed within 1.62% (shown in Table 17). This is because the service delay occurs due to the ATA error model of the aircraft, which happens after flights are assigned their CTAs. Swapping CTAs only changes the relative sequence of flights and does not change the inter-flight spacings in the CTA queue. Thus, the service delay should not be affected by the differences between the two schemes. Within each scheme, we observe the same trend as expected: as the standard deviation of ATA error model noise σ increases, the total service delay cost increases; as AAR increases, the total service delay cost decreases. This is as expected because the AAR here represents the airport landing

capacity and the higher this value is, the lower service delay flights will receive, considering the ATA error model noise stays the same.

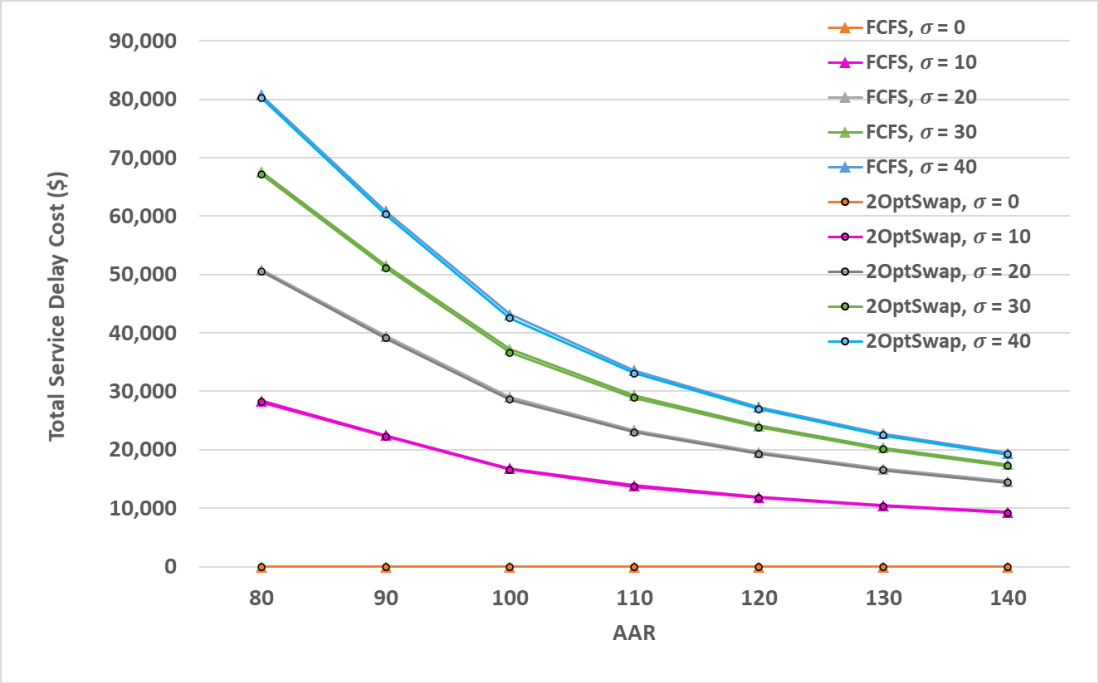

Figure 23 Total Service Delay Cost vs. AAR for Different Noise Models and Scheduling Schemes

Table 17 Total Service Delay Cost vs. AAR for Different Noise Models and Scheduling Schemes ($)

	AAR	80	90	100	110	120	130	140
	Demand/Capacity	0.98	0.87	0.78	0.71	0.65	0.60	0.56
FCFS	sigma = 0	0	0	0	0	0	0	0
	sigma = 10	28,360	22,466	16,833	13,879	11,918	10,480	9,363
	sigma = 20	50,847	39,496	29,041	23,348	19,559	16,750	14,602
	sigma = 30	67,620	51,584	37,195	29,308	24,146	20,356	17,491
	sigma = 40	80,760	60,856	43,163	33,554	27,324	22,787	19,424
2OptSwap	sigma = 0	0	0	0	0	0	0	0
	sigma = 10	28,174	22,252	16,590	13,661	11,743	10,310	9,215
	sigma = 20	50,530	39,136	28,632	22,991	19,280	16,494	14,387
	sigma = 30	67,208	51,121	36,669	28,863	23,814	20,070	17,266
	sigma = 40	80,275	60,318	42,563	33,060	26,974	22,498	19,207
Reduction	sigma = 0	0	0	0	0	0	0	0
	sigma = 10	186	214	243	218	175	170	148
	sigma = 20	317	360	409	357	279	256	214

	sigma = 30	412	464	526	445	332	285	226
	sigma = 40	484	538	600	494	350	289	217
Reduction Percentage	**sigma = 0**	0	0	0	0	0	0	0
	sigma = 10	0.7%	1.0%	1.4%	1.6%	1.5%	1.6%	1.6%
	sigma = 20	0.6%	0.9%	1.4%	1.5%	1.4%	1.5%	1.5%
	sigma = 30	0.6%	0.9%	1.4%	1.5%	1.4%	1.4%	1.3%
	sigma = 40	0.6%	0.9%	1.4%	1.5%	1.3%	1.3%	1.1%

In terms of the total CTA delay cost, there were no differences in the CTA assignment part among the 5 noise models and the total CTA delay cost results are thus identical among the 5 noise models. This is as expected, and the values are presented in Table 18 and Figure 24. A significant reduction for the total CTA delay cost can be observed when switching from FCFS to 2OptSwap. Compared to the FCFS only scheme, our 2OptSwap scheme can reduce the total CTA delay cost by 185 thousand dollars (when AAR = 80 flights/hour) to 11 thousand dollars (when AAR = 140 flights/hour). In terms of the total CTA delay cost reduction percentage, 2OptSwap can achieve 20% to 30% in the reduction percentage, compared to FCFS, under different AAR values. When the AAR is very large, the total CTA delay cost reductions are relatively low because there is not much CTA delay in the system and not many swaps are needed (passing the *CTA delay cost* check). Within our parameter settings, the 2OptSwap scheme generates the most absolute reduction values when the AAR is 80, and this is also when we have the overall highest total CTA delay cost.

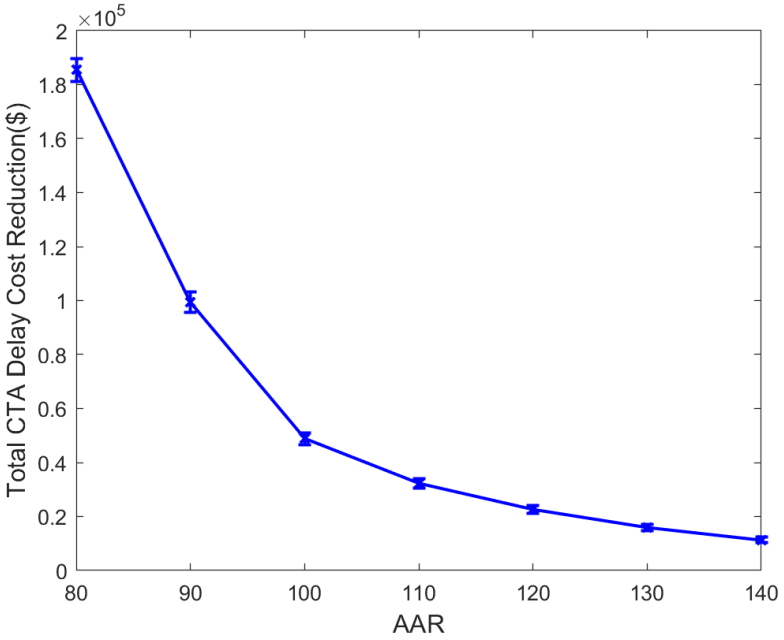

Figure 24. Total CTA Delay Cost Reduction ($) vs. AAR

Table 18 Run 2: Total CTA Delay Cost vs. AAR for Different Scheduling Schemes ($)

AAR	80	90	100	110	120	130	140
Demand/Capacity	0.98	0.87	0.78	0.71	0.65	0.60	0.56
Baseline (FCFS)	947,404	345,865	165,240	113,440	84,884	68,655	57,645
2OptSwap	761,953	246,504	116,414	81,190	62,231	52,781	46,295
Reduction	185,451	99,361	48,826	32,250	22,653	15,874	11,350
Reduction %	19.6%	28.7%	29.5%	28.4%	26.7%	23.1%	19.7%

As a comparison, Monte Carlo simulations comparing FCFS_w/o_ADV and the 2OptSwap based on it are also performed for the same range of AARs and noise parameters. Results were shown in Table 19. It is worth mentioning that although we only changed the advancing rules in the baseline model, the 2OptSwap solution also changed. This is because the 2OptSwap uses the FCFS_w/o_ADV as a starting point and improves the solution based on it. Additionally, the FCFS_w/o_ADV solution is used in the maximum delay constraint for the 2OptSwap. Comparing Table 19 to

Table 18, it can be seen that the 2OptSwap can achieve comparable levels of total CTA delay cost reductions percentages, no matter which baseline scheme we use, FCFS or FCFS_w/o_ADV.

Table 19 Run 2: Total CTA Delay Cost vs. AAR for Different Scheduling Schemes (FCFS_w/o_ADV) ($)

AAR	80	90	100	110	120	130	140
Demand/Capacity	0.98	0.87	0.78	0.71	0.65	0.6	0.56
Baseline (FCFS_w/o_ADV)	961,565	360,939	176,320	123,635	94,585	77,580	65,971
2OptSwap	773,762	257,370	123,435	87,516	68,082	58,118	51,149
Reduction	187,802	103,569	52,885	36,119	26,503	19,462	14,822
Reduction %	19.53%	28.69%	29.99%	29.21%	28.02%	25.09%	22.47%

5.5 Simulations Comparing Different Delay Scenarios

To understand the model performance in different delay scenarios, we simulated flight ETAs from their STAs in "low", "medium (med)" and "high" delay scenarios using the following parameters in Eq. (1) for the 2OptSwap Scheme:

- low: no arrival delay distribution added;
- medium: arrival delay distribution ~ *exp* (5 minutes) added;
- high: arrival delay distribution ~ *exp* (10 minutes) added.

From Figure 25, it can be seen that the total delay cost under 2OptSwap (dashed lines in the figure) is always lower than under the FCFS assignment scheme (solid lines in the figure) for the same parameter settings. For each AAR, the delay cost is significantly different for different ETA delay scenarios. For each delay scenario, the delay cost decreases as the AAR increases, and all scenarios follow similar trends.

This is expected because the differences in delay scenarios are really tantamount to different amounts of time shifts, which translates to shifts of delay costs relative to STAs.

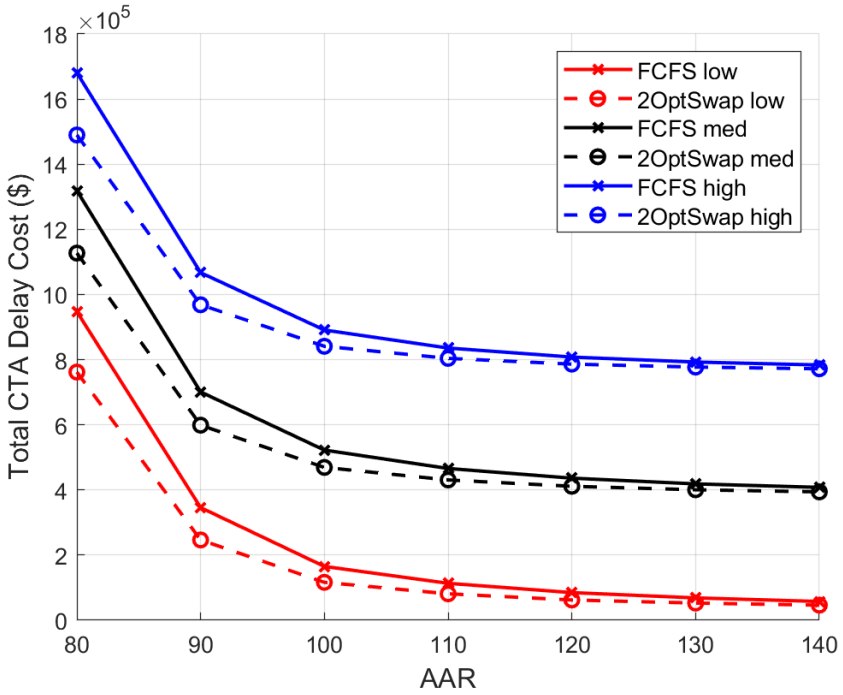

Figure 25 Total CTA Delay Cost for 3 Departure Delay Scenarios under 2 Schemes

Figure 26 presents the total delay cost reduction by using 2OptSwap compared to using the FCFS scheme. The same trend as in Figure 24 is observed for each delay scenario and no significant difference among the three delay scenarios.

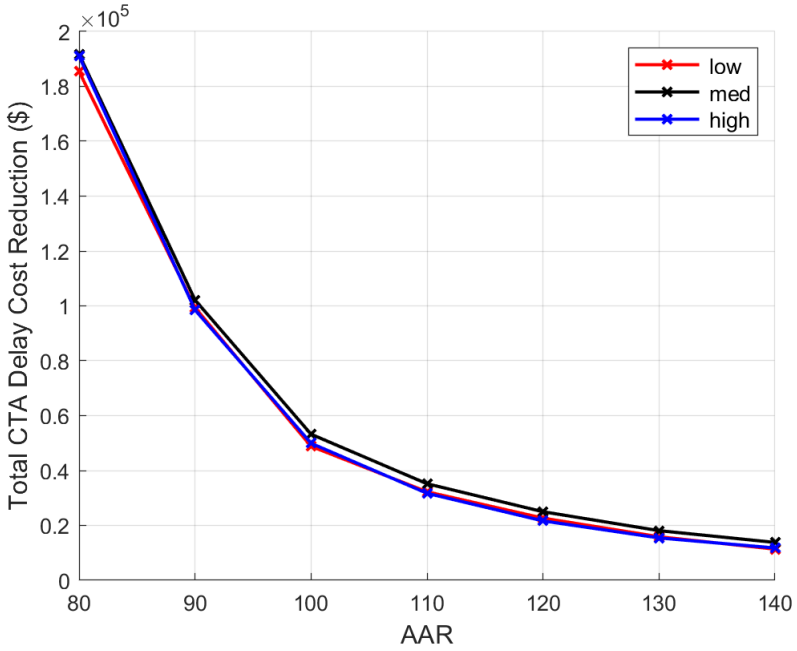

Figure 26 Total CTA Delay Cost Reduction from FCFS to 2OptSwap in 3 Delay Scenarios

Figure 27 presents the total CTA delay cost reduction percentage achieved from 2OptSwap compared to the FCFS scheme. The reduction percentage values for the low delay scenario are the largest among all scenarios because the denominators (delay cost based on FCFS solution) are much smaller. Similarly, the reduction percentage values for the medium delay scenario are larger than for the high delay scenario.

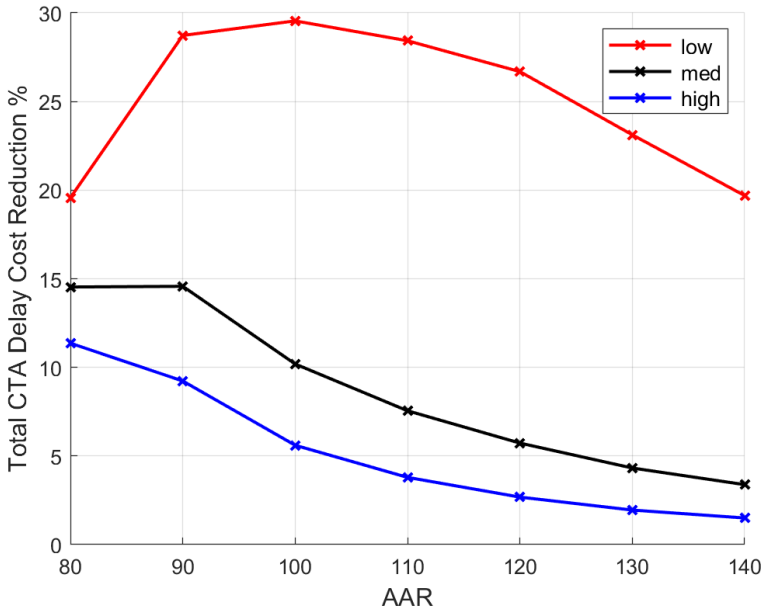

Figure 27 Total Delay Cost Reduction % from FCFS to 2OptSwap in 3 Delay Scenarios

5.6 Simulations Exploring Fairness Among Carriers

Another set of simulations was conducted to explore the model performance under two different carrier distributions:

1. Use the uneven carrier distribution as-is. There are 36 carriers in our dataset. The carrier distribution was described in Section 4.2.1.

2. Create 10 "fictitious carriers" and randomly distribute the 1302 flights into relatively equal shares of fictitious carriers. Carrier A, Carrier B, …Carrier H each have 130 flights, and Carrier I and Carrier J each have 131 flights. The same percentage of connecting passengers is used for all flights in different carriers.

Similarly, three sets of ETA delay parameters are used to represent low, medium, and high delay scenarios.

Within the three simulations (low, medium, and high delay scenarios) using fictitious carriers, we observe similar trends as in the three simulations using real carriers (discussed in Section 5.5).

Within each delay scenario, it is observed that the delay cost reduction values for the fictitious carriers are always slightly higher than the values for the real carriers. This could indicate that our model performs better when the flights are more evenly distributed among all carriers. This provides a hint as to how model results might change when applied to a different airport with a different distribution of market concentration.

Within each carrier, we track the CTA delay cost in FCFS, the CTA delay cost in 2OptSwap, and calculate the total and average CTA delay cost reduction of switching from FCFS to 2OptSwap per flight. The average CTA delay cost reduction is a more reasonable metric than the reduction percentage when the comparison is performed among different carriers, since the latter could be affected by the different scales of denominators. The previous results obtained for the aggregate of all carriers (in Section 5.5) revealed that the total CTA delay cost reduction values differ in the 21 different scenarios (7 AAR values $\times$ 3 delay scenarios, shown in Figure 26). It was thus not surprising to find these variations within each carrier for different scenarios

as well. We thus plot boxplots to represent this variation within each carrier in Figure 28 and Figure 30.

However, when comparing among different carriers, the distribution of these 21 average delay cost reduction values for different carriers are very different amongst each other (shown as different boxplots among different carriers in Figure 28 and Figure 30).

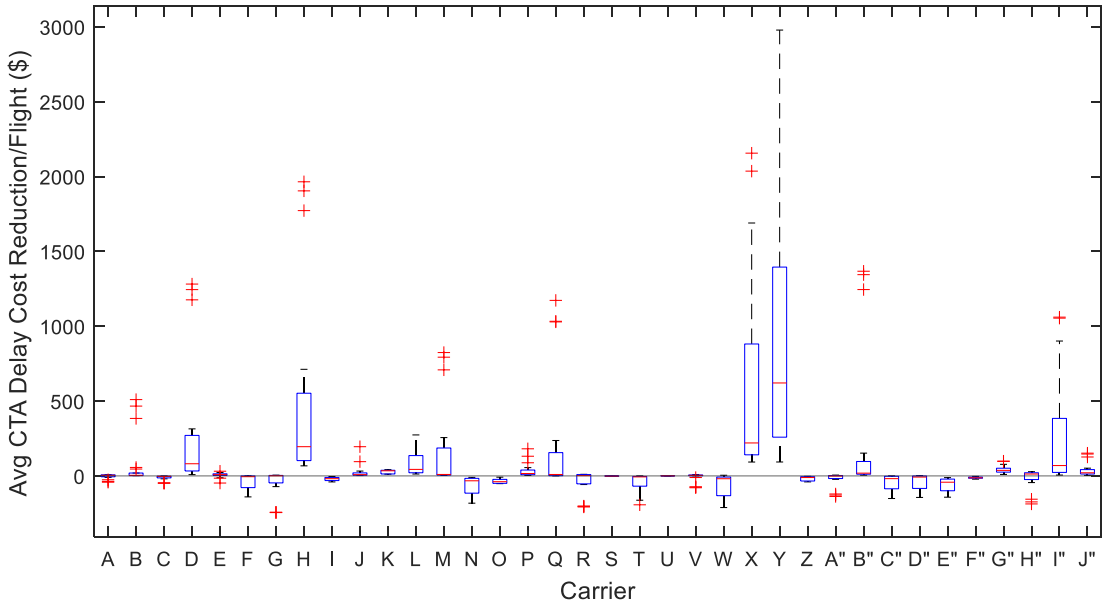

Figure 28 The Average Delay Cost Reduction for All Real Carriers

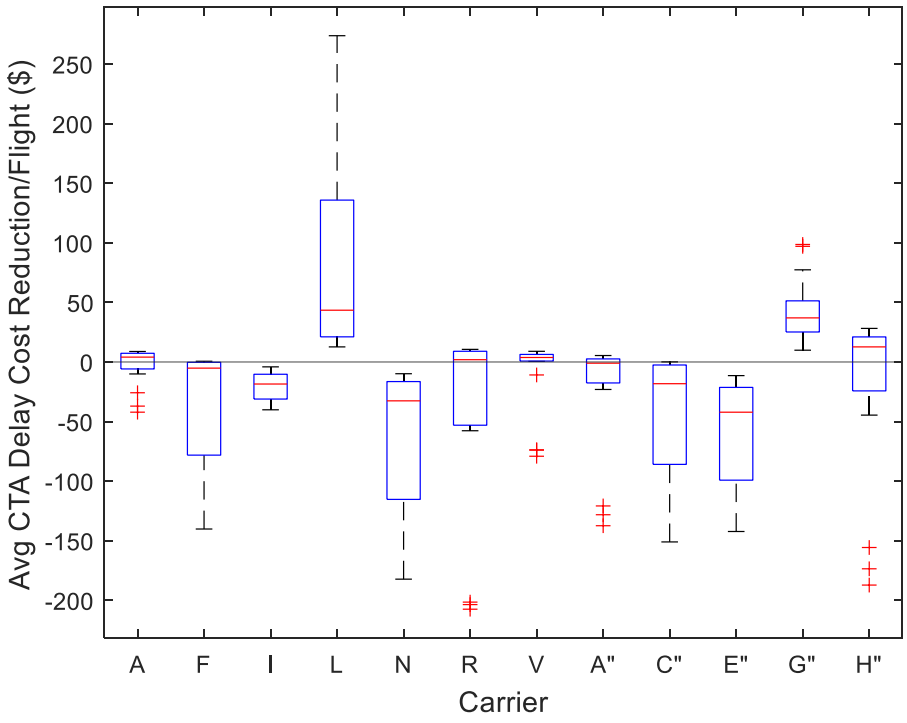

Figure 29 The Average CTA Delay Cost Reduction among Large Real Carriers

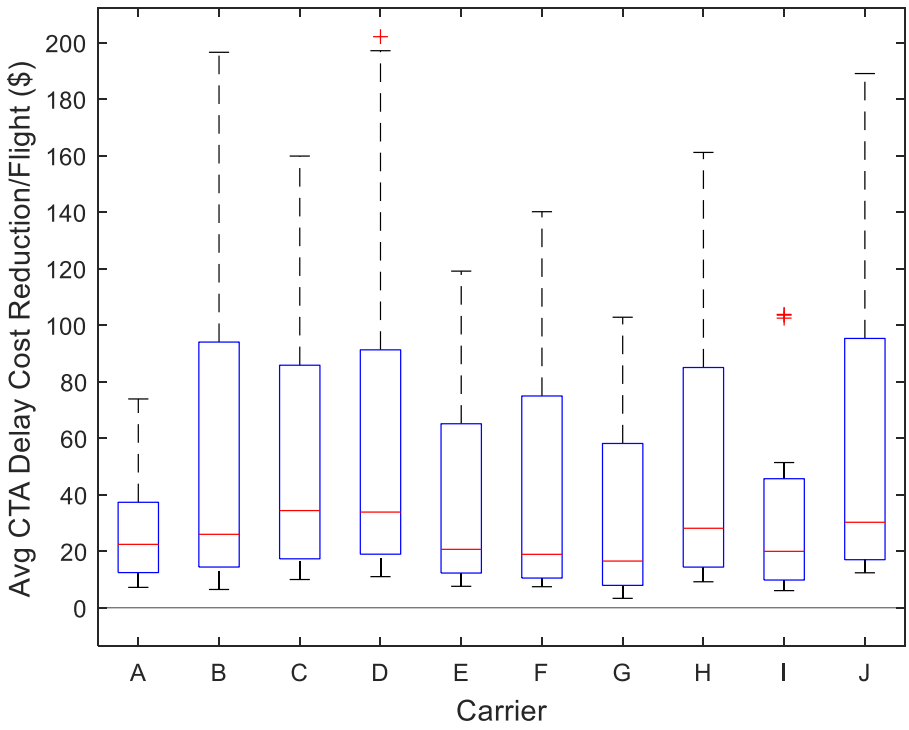

Figure 30 Delay Cost Reduction % for 10 Fictitious Carriers

In Figure 28, the real carrier names are anonymized as Carrier A, Carrier B, …, for sensitivity reasons. It is observed that the average CTA delay cost reduction values are unevenly distributed among different carriers. Some carriers tend to have larger average CTA delay cost reduction values while some carriers have smaller or even negative average CTA delay cost reduction values. For those carriers with extremely high average reduction values (H, X, and Y in Figure 28), further investigation reveals they are carriers that only have 1 flight within the day, and they are all international flights that have very large aircraft sizes and numbers of seats. This is understandable since our 2OptSwap identifies them as high-cost flights and swaps them forward, incurring large average delay cost reduction values. To exclude such

extreme cases where carriers have very small numbers of flights, we selected carriers that have more than 5 flights during the day and plotted them as “large carriers” in Figure 29. A closer look into carriers that tend to have small or negative reduction values reveals that the sizes of aircraft (represented by average numbers of seats per aircraft) in these carriers are often smaller than the sizes of aircraft in those carriers that tend to have large average delay cost reduction values.

Even for those fictitious carriers that have almost even numbers of flights (shown in Figure 30), this average CTA delay cost reduction values among different carriers is still uneven, but this distribution is much less uneven than those in real carriers (Figure 28 and Figure 29) and there are no negative values in these fictitious carriers. This is as expected because the fictitious carriers are assigned completely at random among all the flights, the size and average numbers of seats per aircraft within carriers are approximately equal among all the fictitious carriers, and we used the same parameter value for the fraction of connecting passengers assigned to all the fictitious carriers. While in real carrier distributions, the number of aircraft within a carrier ranges between 1 and 811, the average number of seats per aircraft within carriers, and the parameters used for generating connecting passenger percentage are all quite different.

In short, the observed uneven distribution of average CTA delay cost reduction among different carriers could originate from different distributions of number of seats. Carriers that operate larger aircraft have an advantage because the delay costs of their aircraft (calculated using number of non-connecting passengers, number of

connecting passengers, and others) are usually larger than those of smaller aircraft from other carriers. Therefore, the algorithm tends to swap larger aircraft forward because this reduces the total CTA delay cost over all flights. Another factor responsible for the uneven results is for carriers that have only a very small number of flights throughout the day, the average delay cost reduction could be volatile since it is based solely on the performance of a very limited number of flights. These findings reveal that improvements could be made to provide equal benefits to carriers with small aircraft and/or that operate relatively few flights.

To encourage all carriers to participate in this new CTA assignment scheme, a post assignment adjustment could be considered that makes sure all carriers gain the same amount of benefits from participating. A simple compensation scheme could be at the end of the day, we match the average delay cost reduction for all carriers to be the same value as the average delay cost reduction for all flights among all carriers. For example, if the average delay cost reduction value is $50 for all flights among all carriers, those carriers that achieved reduction values higher than $50 will pay the additional amount and those carriers that achieved reduction values lower than $50 will receive compensation for the lacking amount, so all carriers achieve equal amount of average delay cost reduction. Similar adjustments have been proposed in [20] where side payments are suggested to compensate carriers operating smaller aircraft for absorbing increases in delay. Even though such adjustments are challenging politically and bring complexity in implementation, they are consistent with moves toward market mechanisms on the part of the air traffic management

community [20, 60]. Adjustments to the 2OptSwap could also be implemented to achieve a more even performance distribution but this may limit the total amount of delay cost reduction we could achieve for all flights.

In later sections (Section 6.4 and Section 7.4), a fairness metric and the corresponding fairness adjustments under a dynamic scheme are introduced to better handle the uneven fairness distribution issue.

5.7 Simulations Exploring Extended Metering

In the previous simulations, we explored a CTA assignment scheme when flights cross the freeze horizon, and the distance of the freeze horizon from the metering fix at the destination airport is set at 250 NM.

Meanwhile, TBFM has the concept of "extended metering" which starts metering aircraft well in advance in the en route environment and it has already been implemented in four locations [61]. Similarly, this study explores extending the decision-making point of our CTA assignment scheme further away.

In this study, for short range flights (with origins within the decision-making distance), it is assumed that they can be entered into the TBFM system early, and the CTA assignment system could assign CTAs to them before their departure. Such an idea has been tested and proved to reduce the delays for short-range flights [62]. Under this assumption, extending the decision point to a further distance is essentially

a proxy for relaxing the maximum advancing time constraint in Eq.(2). Additional studies could test scenarios where this "early entering" is not allowed and integrate GDP into this TBFM system since ground delay costs are different from airborne delay costs, but such cases are beyond the scope of this dissertation and the corresponding unfairness between short-range flights and long-range flights could be challenging to justify.

A set of different decision point distances are tested: 250 NM, 400 NM, 600 NM, 800 NM, and 1000 NM, for the FCFS and 2OptSwap schemes. For each distance setting, simulations are conducted under 7 AAR values in [80, 90, 100, 110, 120, 130, 140] flights/hour and 500 Monte Carlo realizations. The total CTA delay cost is tracked and shown in Table 20 and Figure 31. In Figure 31, each subplot represents an AAR value, and each subplot has different ranges on the y-axis. It can be seen that given a fixed AAR value, as the decision point distance (on the x-axis) increases, the total CTA delay cost decreases for both FCFS (blue) and 2OptSwap (red) schemes. This is as expected because extending the decision point means flights will have more feasibilities to be advanced, thus incurring less delay and the corresponding delay cost. Given a fixed decision point distance, as the AAR value increases (from one subplot to another), the airport capacity increases, and the total CTA delay cost decreases. This is as expected and the same as in previous simulations.

Table 20 Total CTA Delay Cost ($) vs. Decision Point Distance (NM) (FCFS vs. 2OptSwap)

	Distance (NM)	250	400	600	800	1000
AAR = 80	FCFS	947404	923546	899786	878945	856909
	2OptSwap	761953	740203	718818	700178	680611
	reduction	185451	183343	180968	178767	176298

AAR = 90	reduction %	19.57%	19.85%	20.11%	20.34%	20.57%
	FCFS	345865	326304	306951	290510	274427
	2OptSwap	246504	230052	214165	200822	188029
	reduction	99361	96251	92786	89688	86398
	reduction %	28.73%	29.50%	30.23%	30.87%	31.48%
AAR = 100	FCFS	165240	152927	139687	129217	120732
	2OptSwap	116414	105820	94868	86308	79561
	reduction	48826	47107	44818	42909	41172
	reduction %	29.55%	30.80%	32.08%	33.21%	34.10%
AAR = 110	FCFS	113440	103721	94547	87217	81951
	2OptSwap	81190	72520	64371	58097	53608
	reduction	32250	31201	30176	29120	28343
	reduction %	28.43%	30.08%	31.92%	33.39%	34.59%
AAR = 120	FCFS	84884	77293	70186	65217	61530
	2OptSwap	62231	55161	48512	43842	40427
	reduction	22653	22132	21674	21374	21103
	reduction %	26.69%	28.63%	30.88%	32.77%	34.30%
AAR = 130	FCFS	68655	62549	57317	53682	51131
	2OptSwap	52781	46702	41305	37515	34804
	reduction	15874	15847	16012	16167	16327
	reduction %	23.12%	25.34%	27.94%	30.12%	31.93%
AAR = 140	FCFS	57645	52816	48850	46249	44486
	2OptSwap	46295	41127	36608	33527	31305
	reduction	11350	11689	12241	12722	13182
	reduction %	19.69%	22.13%	25.06%	27.51%	29.63%

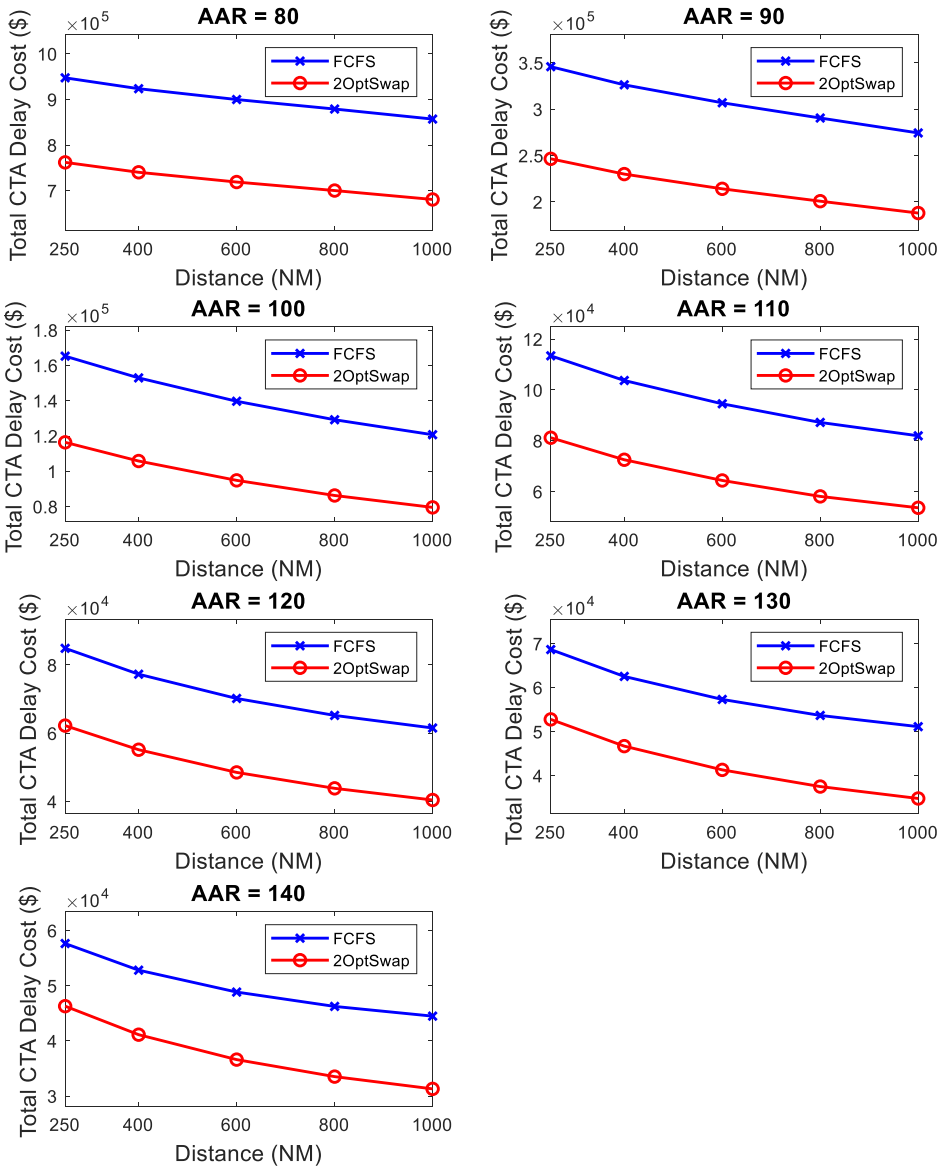

Figure 31 Total CTA Delay Cost ($) vs. Decision Point Distance (NM) (FCFS vs. 2OptSwap)

As a comparison, a set of simulation experiments using FCFS_w/o_ADV as the baseline scheme are also conducted. In this baseline model, flights are not able to be advanced in terms of CTA slots. This set of simulations are controlled in that all other

parameters and settings are exactly the same as the previous set of simulations in this section. The results are shown in Table 21 and Figure 32 below. It can be seen that the total CTA delay cost under FCFS_w/o_ADV scheme (blue) decreases very slowly as the decision point distance increases. In fact, the curves stay almost flat in the plot in Figure 32. This is as expected because under this scheme, no advancing is allowed in terms of CTA slots. Increasing the decision point distance only slightly affects some CTA slots' feasibility to the flights. However, under this 2OptSwap scheme (red), the CTA total delay cost decreases much faster than in FCFS_w/o_ADV scheme (blue), and slower than the 2OptSwap curves (red) in Figure 31, as the decision point distance increases. This is as expected because the 2OptSwap uses the solutions in FCFS_w/o_ADV as a starting point and it allows flights being swapped forward to advance flights. This also naturally leads to the increase in total CTA delay cost reduction values from FCFS_w/o_ADV to 2OptSwap, as the decision point distance increases (shown as the gap between the two curves in Figure 32 and the row "reduction" in Table 21).

Table 21 Total CTA Delay Cost ($) vs. Decision Point Distance (NM) (FCFS_w/o_ADV vs. 2OptSwap)

	Distance (NM)	250	400	600	800	1000
AAR = 80	FCFS w/o_ADV	961565	957800	955812	955165	954827
	2OptSwap	773762	769046	765975	764502	763479
	reduction	187802	188753	189837	190663	191348
	reduction %	19.53%	19.71%	19.86%	19.96%	20.04%
AAR = 90	FCFS w/o_ADV	360939	357544	356247	355828	355632
	2OptSwap	257370	252583	249530	247810	246619
	reduction	103569	104960	106717	108018	109014
	reduction %	28.69%	29.36%	29.96%	30.36%	30.65%

AAR = 100	**FCFS w/o_ADV**	176320	174507	173718	173415	173274
	2OptSwap	123435	119643	116778	114966	113643
	reduction	52885	54864	56940	58449	59631
	reduction %	29.99%	31.44%	32.78%	33.70%	34.41%
AAR = 110	**FCFS w/o_ADV**	123635	122303	121798	121561	121445
	2OptSwap	87516	84152	81473	79671	78371
	reduction	36119	38151	40325	41890	43073
	reduction %	29.21%	31.19%	33.11%	34.46%	35.47%
AAR = 120	**FCFS w/o_ADV**	94585	93627	93228	93049	92966
	2OptSwap	68082	65031	62400	60650	59332
	reduction	26503	28595	30828	32399	33634
	reduction %	28.02%	30.54%	33.07%	34.82%	36.18%
AAR = 130	**FCFS w/o_ADV**	77580	76837	76515	76381	76314
	2OptSwap	58118	55289	52796	51112	49816
	reduction	19462	21548	23718	25270	26498
	reduction %	25.09%	28.04%	31.00%	33.08%	34.72%
AAR = 140	**FCFS w/o_ADV**	65971	65364	65120	65011	64948
	2OptSwap	51149	48480	46077	44382	43084
	reduction	14822	16884	19043	20629	21863
	reduction %	22.47%	25.83%	29.24%	31.73%	33.66%

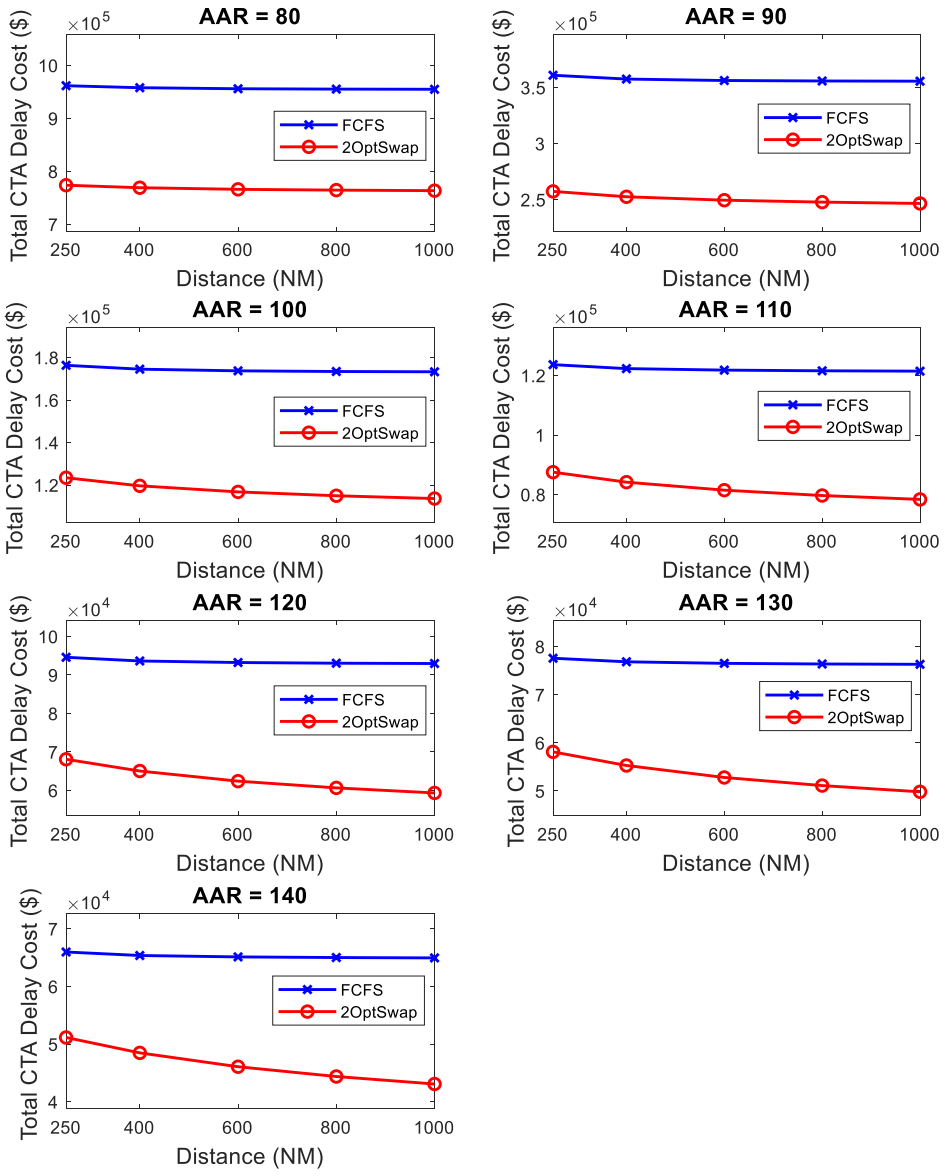

Figure 32 Total CTA Delay Cost ($) vs. Decision Point Distance (NM) (FCFS_w/o_ADV vs. 2OptSwap)

5.8 Chapter Conclusion

In this chapter, simulations for four different CTA assignment schemes, the FCFS baseline scheme, the IP optimization scheme, the 2OptSwap scheme, and the 2OptSwap with simulated annealing scheme, were designed and performed for a full day period of arrival flights in ATL. An alternative baseline model, FCFS_w/o_ADV, was also tested. Real historical flights data were used, and parameters were varied for different sets of simulations. For each parameter setting, 500 Monte Carlo realizations were conducted where slightly different flight information was generated to use as inputs.

First, in Section 5.1, two different baseline schemes, FCFS and FCFS_w/o_ADV, were compared. The former allows advancing while the latter does not. It was observed that allowing advancing could decrease the total CTA delay cost in FCFS from 1.47% to 12.62%, compared to FCFS_w/o_ADV, under different AAR values. FCFS was kept as the major baseline model for the following simulations.

In Section 5.2, different neighbor searching sizes in the 2OptSwap scheme were explored and a value of 20 was selected to use for the succeeding simulations based on the tradeoff between the total CTA delay cost and the runtime. The carrier preference check was removed from the 2OptSwap, and the priority penalty was removed from the objective function in the IP optimization to make results comparable. Following this setting, in Section 5.3, the performance among the four CTA assignment schemes were performed and 2OptSwap was chosen over

2OptSwap_SA and IP optimization model for the succeeding case studies, due to compatible solution quality but much shorter runtime.

In Section 5.4, the carrier preference check was added back into 2OptSwap and the performance in terms of the total service delay cost and the total CTA delay cost between 2OptSwap and FCFS schemes was compared. There were no significant differences in the total service delay cost among the two CTA assignment schemes but the 2OptSwap could provide a 20% to 30% reduction from the FCFS baseline scheme in terms of total CTA delay cost.

In Section 5.5, simulations were conducted to evaluate the model performance under three different delay scenarios. Flights' ETAs were generated from their STAs in "low", "medium" and "high" three delay scenarios. Results show that the total delay cost reduction remains relatively similar among different ETA delay scenarios, which means the swapping mechanism could potentially realize similar levels of cost savings regardless of whether the incoming flights are behind their schedules or not. In relative terms, the delay cost reduction percentage (relative to the FCFS baseline) tends to be higher for lower delay scenarios, which reflects the fact that lower delay scenarios have lower total delay costs.

In Section 5.6, simulations were conducted to evaluate one aspect of fairness among different carriers by testing two carrier distributions. The distribution of the average CTA delay cost reduction among different carriers could be very different, depending on carrier characteristics (aircraft size, number of passengers, relative frequency of

flights, etc.). These results show potential challenges to establish fairness. A post assignment adjustment is proposed to match all carriers with the same amount of delay cost reduction. Inter-carrier swapping rules could also be modified to apply different weights (to compensate for flight frequencies and/or aircraft sizes). This dissertation further explores approaches for fairness adjustments in Section 6.4 and Section 7.4.

In Section 5.7, we explored extending the distance of the decision-making point from 250 NM to 400 NM, 600 NM, 800 NM, and 1000 NM. Extending this distance reduces the total CTA delay cost since it allows the flights to be advanced more. We also tested the same parameter settings under a FCFS_w/o_ADV baseline model and the total CTA delay cost under this model does not reduce much as the decision-making point distance increases.

Chapter 6: The Dynamic System Design and Real-Time Metrics

6.1 Incentives for a Dynamic System

In Chapter 3 and Chapter 4, approaches to tackle the flight CTA assignment problem on a daily basis were presented. In these previous models, parameters such as AARs remained static throughout the day. However, in real situations, we do not have a whole day's information on hand at its outset, and the environment is constantly changing. In this chapter, the rolling horizon (RH) concept is integrated into the simulation process to form a dynamic system. That is, rather than solving the problem considering the whole day time frame, we solve it for successive planning intervals representing a small part of the day, reducing the size of the problem per interval [63] so that the system is able to dynamically capture any new changes and perform assignment periodically in relatively short computation time based on the new information available. In the literature, this rolling horizon concept has been applied and tested in several scheduling problems under uncertainty [64, 65].

In this chapter, the CTA assignment problem re-runs during each period (epoch). At the beginning of each epoch, the system includes any new information such as the flights' ETAs and the available CTA slots, and updates any existing information, e.g., airport AARs, wind characteristics for noise generations, and so on. The AARs are used to calculate new CTA slots for the new time period, and the wind characteristics are used in generating noise to simulate flight ATAs. This is extremely useful on

unpredicted disturbances such as runway closures and turbulences. For example, instead of using one fixed AAR value throughout the day, the values of AARs could vary in each time interval to simulate realistic capacity change scenarios.

Another benefit of using a dynamic scheme is that it allows the usage of real-time metrics. Real-time metrics can serve as feedback from previous actions and be used to adjust future actions based on previous performance.

As specified above, the rolling horizon approach can be seen as a timewise decomposition technique and is not confined to any specific scheduling scheme. It can be applied to any existing CTA assignment scheme, such as the FCFS, 2OptSwap, 2OptSwap_SA, and the IP optimization model.

6.2 The Rolling Horizon Approach Design for CTA Assignment

A rolling horizon approach is proposed to perform the CTA assignment dynamically. This section describes the design and major steps in this approach. The rolling horizon approach divides the time of interest into smaller consecutive rescheduling intervals. At the beginning of each interval, we update the incoming flights set and available CTA slots and solve the CTA assignment given the most recent parameters. In addition to this rescheduling interval, a look-ahead window is also needed to include the data from the close future period to provide a better solution in the current period.

There are three major steps in this rolling horizon approach, and they are described below.

a) Initialize Datasets and Assign CTAs

This step occurs at the beginning of the rolling horizon loop at the beginning time t_0. A simple illustration is provided in Figure 33. A look-ahead time window t_l is used to select incoming flights that have ETAs within the range $[t_0,\ t_0\ +\ t_l)$. We then assign CTAs to these flights. Depending on the CTA assignment scheme, these flights might be assigned CTA slots in the same sequence as their ETAs or differently. Some flights might receive CTAs earlier than their ETAs if speed-up is within limits and the assignment scheme allows. Depending on the capacity and demand situations, flights may also receive CTAs later than their ETAs. The values for the look-ahead time window and the rescheduling interval should be carefully chosen such that enough information is included in each interval to make the decision without affecting the quality of the overall results [64, 66].

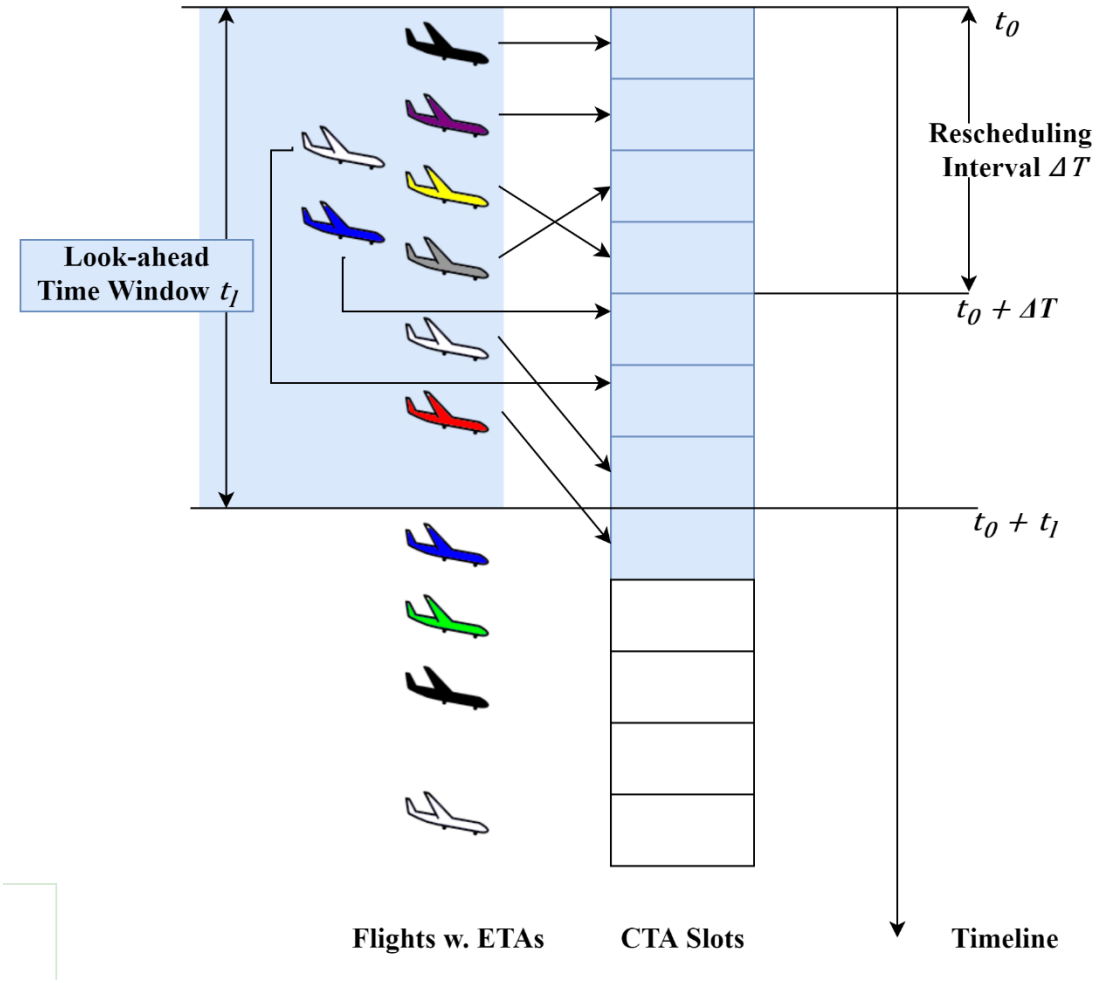

Figure 33 Step a) in the Rolling Horizon Approach

b) Lock CTA - Flight Pairs

In this step, a rescheduling interval ΔT (where $\Delta T < t_l$), also called the horizon size/stepping size, is used to select CTA slots to be locked with certain flights in this epoch (shown in Figure 34). Given the CTA assignments achieved from Step a), we lock all the CTAs within the range $[t_0, t_0 + \Delta T)$ and discard the rest of the CTA assignments. Those flights that already have locked CTA slots are later removed from

the incoming flight list and put into a list of flights to enter the approaching and landing service simulator for the destination airport.

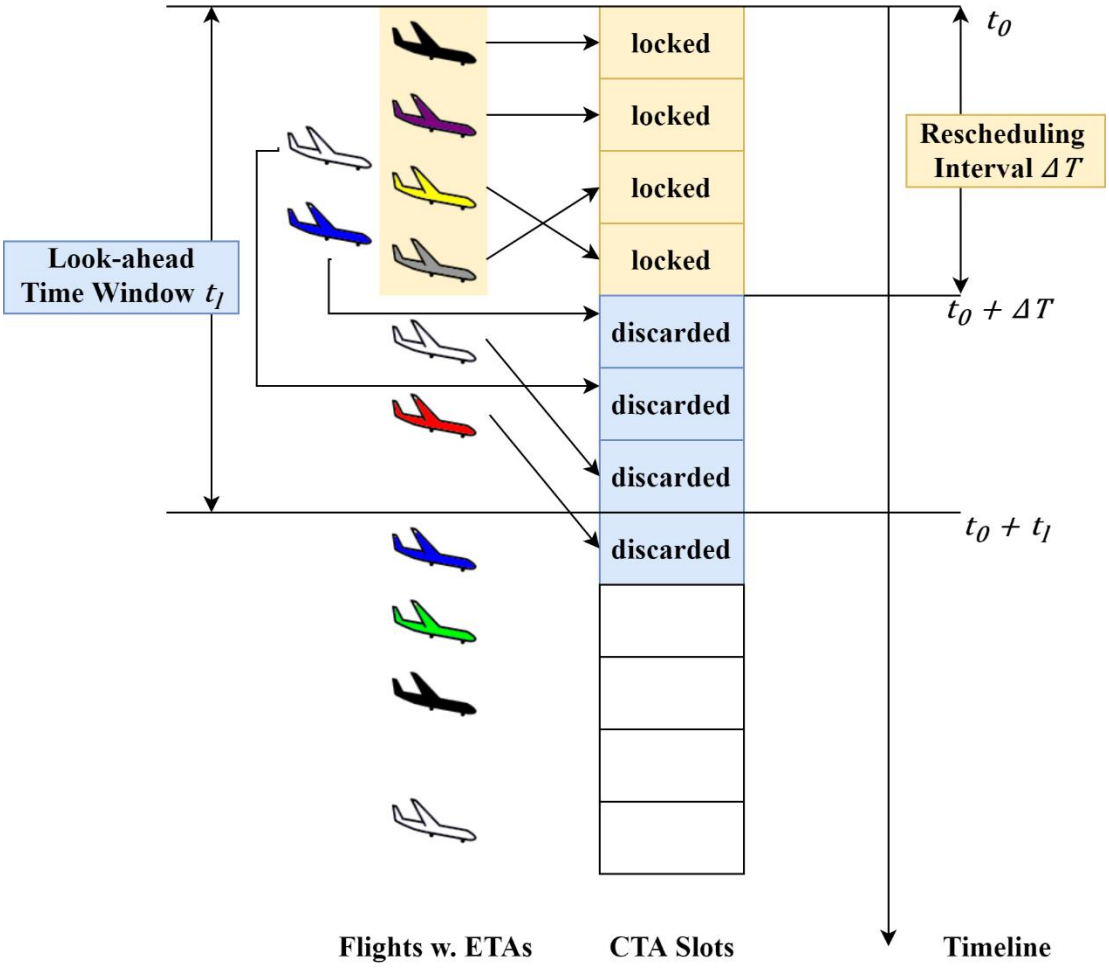

Figure 34 Step b) in the Rolling Horizon Approach

c) Update Datasets and Assign CTAs

In Step a), we showed how a rolling horizon starts with an initial time point. In this section, we illustrate how the rolling horizon "rolls" to the next interval and how the system updates the datasets at the beginning of each following interval.

In this step, the system updates the datasets from previous intervals. The updated datasets should cover any new flights and new information in the system and include any left-over flights that could not be assigned in the previous time intervals. This process is illustrated in Figure 35. It can be seen that those flights already have locked CTAs have been removed from the list. The current time t is updated by adding the rescheduling interval to it ($t = t_0 + \Delta t$), and the first CTA slot at the new interval is set as t or the first available CTA slot time, whichever is greater. The latter guarantees that the system does not over-schedule flights between different epochs. For example, if the last flight in the previous epoch (assuming the previous epoch started at 8 AM) was assigned a CTA of 8:59 AM, our CTA slot length is 5 minutes, and the rescheduling interval is 1 hour, then the next epoch and the next feasible CTA slot will start at 9: 04 AM, instead of 9 AM. The flight list is updated by selecting flights within the look-ahead window and including any leftover flights from previous time intervals. We then assign CTAs to these flights in the new flights list. The rolling horizon loop repeats Step b) and Step c) until it reaches the end time.

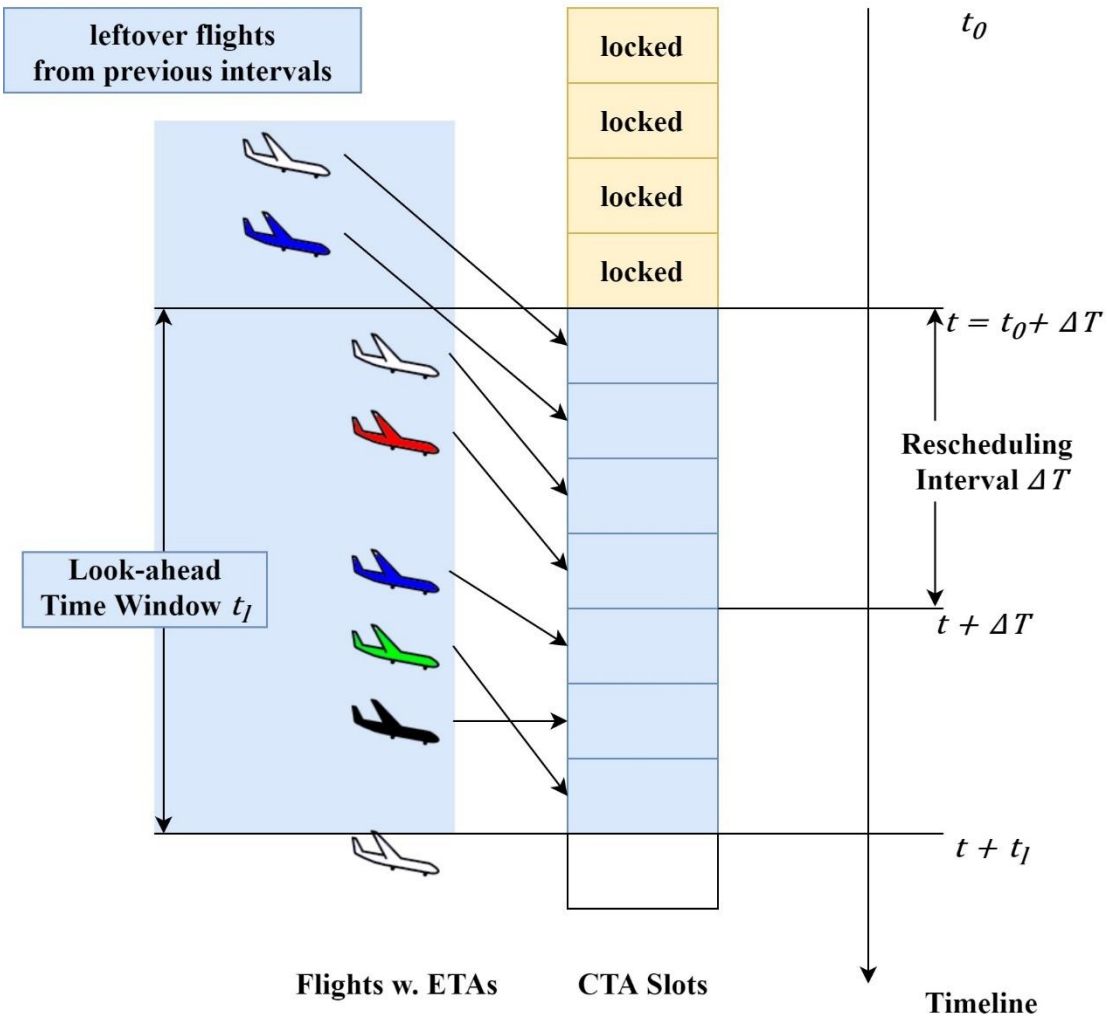

Figure 35 Step c) in the Rolling Horizon Approach

6.3 The Rolling Horizon Approach Design for Flights Landing Service

Similar to the CTA assignment process, the flights landing service can also be implemented under a rolling horizon approach.

After flights are assigned CTAs, they cross the freeze horizon and keep approaching their destination airport. These flights eventually arrive at the metering fix at the

destination and have ATAs the same as or different from their assigned CTAs, depending on their operational situations and ATA error models used along the route. This process is shown as the yellow and green parts in Figure 36.

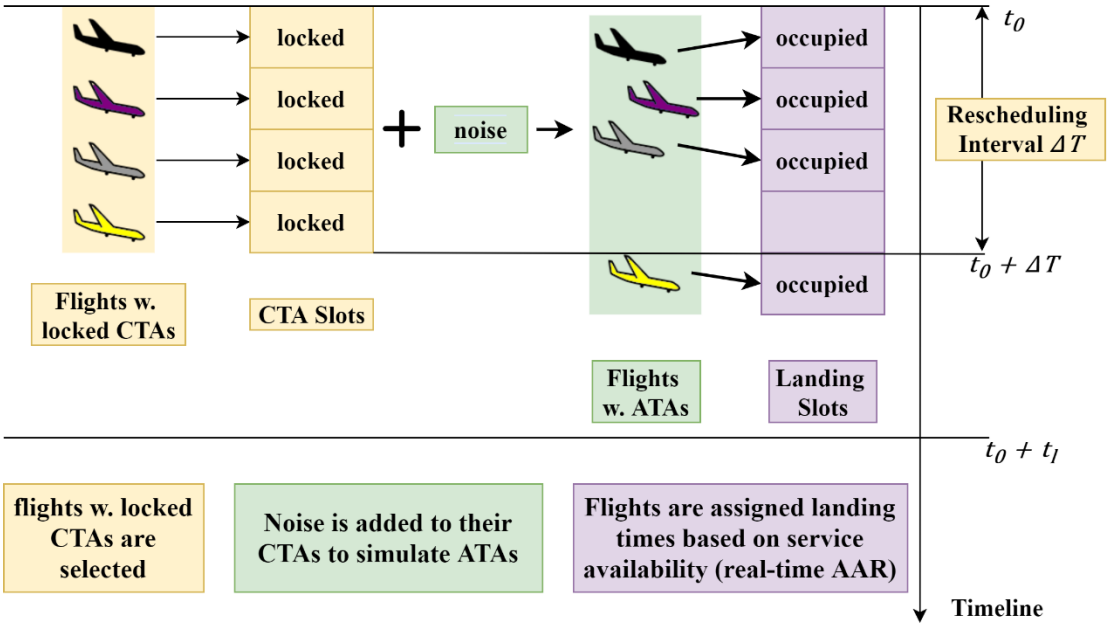

Figure 36 The Landing Service in the Rolling Horizon Approach

When flights arrive at the airport, they will be assigned final landing times based on the runway availability. This process is referred to as the landing service (shown as the purple part in Figure 36) and it also follows a rolling horizon approach. Similar to the steps in Section 6.2, there are three major steps in this approach: a) Initialize Datasets and Assign Landing Slots; b) Lock Landing Slot – Flight Pairs; c) Update Datasets and Assign Landing Slots. The details in each step follow the same principles as in Section 6.2.

The flow chart for this rolling horizon approach in Section 6.2 and Section 6.3 is given in Figure 37. For the same flight, it enters the CTA assignment process before entering the landing service process. This time interval shift between CTA assignment and landing service represents that it takes time for flights to arrive at the airport after the CTAs are assigned, some noise and unexpected disturbance could happen, and the landing service should use the most recent information.

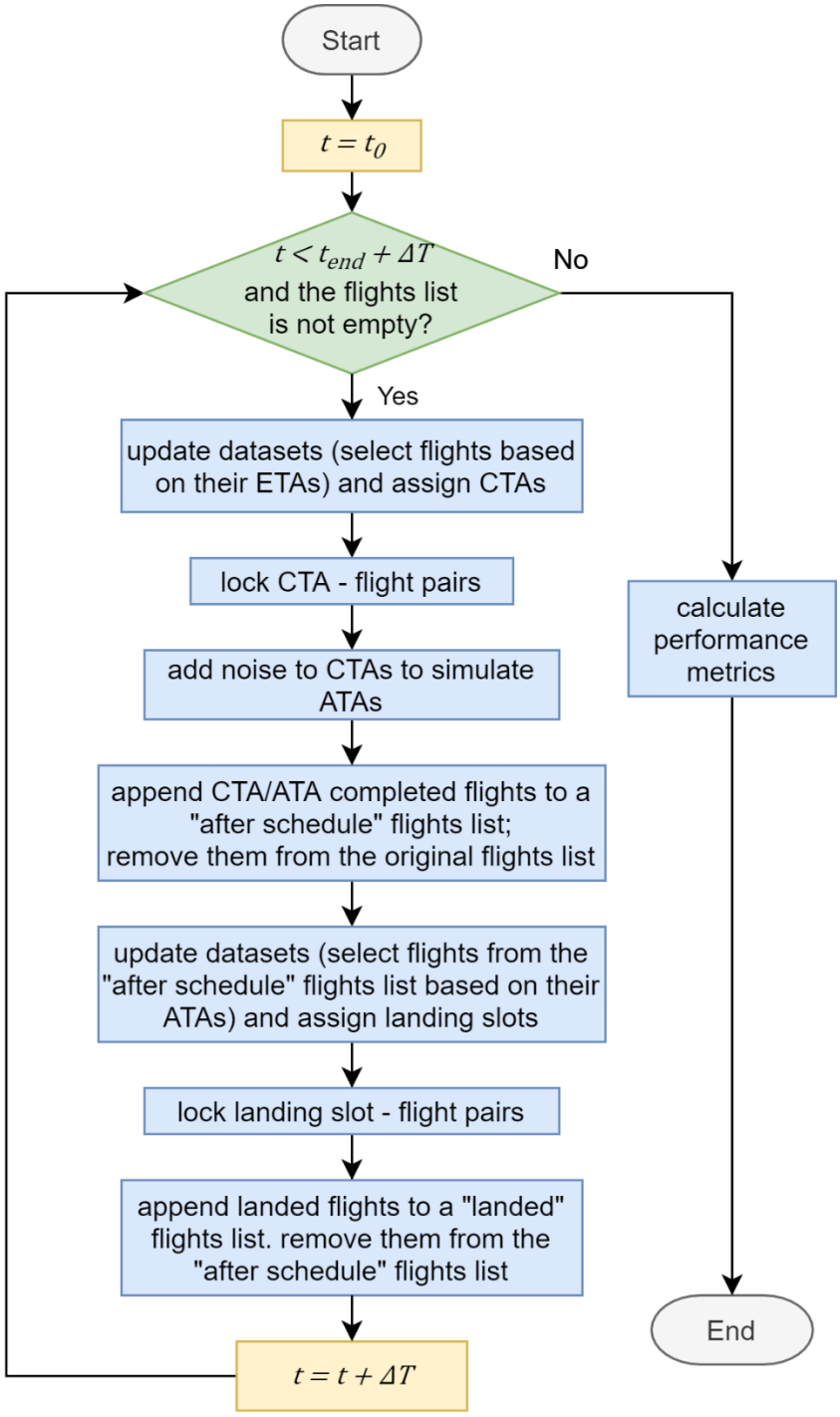

Figure 37 Flow Chart for the Rolling Horizon Approach

6.4 Fairness Metric and its Use in the Rolling Horizon Approach

In this section, we define a fairness metric that attempts to characterize the fairness of carriers' benefits of switching from FCFS to 2OptSwap (or any other scheme of interest). The purpose of defining such a fairness metric is to better investigate the fairness of switching the CTA scheme from FCFS to some other scheme among different carriers.

The fairness metric is built based on the concept of the mean absolute deviation (MAD). Such a fairness metric has been used in existing literatures [67, 68]. The metric can be calculated as below:

- Step 1: For each carrier i, calculate the benefits of switching from FCFS to 2OptSwap (defined by the average CTA delay cost reduction per flight from FCFS to 2OptSwap). The calculation is formulated as Eq.(15). The philosophy behind this benefits definition is that the current FCFS scheme is accepted by all parties and can be assumed as a fair scheme among all carriers. Evaluating the fairness of 2OptSwap among carriers is equivalent to evaluating the fairness of the changes that 2OptSwap brought, compared to the FCFS solution.

$$b_i = \frac{cost\_FCFS_i - cost\_2OptSwap_i}{nFlts_i} \tag{15}$$

where b_i is the benefits of carrier i by switching from FCFS to 2OptSwap, $cost\_FCFS_i$ is the total CTA delay cost in FCFS scheme for all flights in

carrier i, $cost\_2OptSwap_i$ is the total CTA delay cost in 2OptSwap scheme for all flights carrier i, and $nFlts_i$ is the total number of flights for carrier i. This denominator could also include some weight coefficient to capture other information such as aircraft sizes, types, load factors, etc.

- Step 2: Calculate the mean benefits over all carriers, denoted by $\overline{b}$.

- Step 3: For each carrier i, calculate how far away b_i is from $\overline{b}$ using absolute deviation in Eq.(16).

$$|dev_i| = |b_i - \overline{b}|. \tag{16}$$

  where dev_i is the deviation of carrier i's benefits from the mean benefits value over all carriers and $|dev_i|$ is its absolute value, b_i and $\overline{b}$ are defined and calculated in Step 1 and Step 2 above.

- Step 4: Average these absolute deviations over all carriers by Eq.(17).

$$MAD = \frac{\sum_i |b_i - \overline{b}|}{nCarriers} \tag{17}$$

  where $nCarriers$ is the number of carriers in the dataset.

Following the definitions above, a large MAD indicates large benefits deviations from the mean, which can be interpreted as unfairness among different carriers. Conversely, a small MAD indicates small benefits deviations from the mean, meaning that at least according to this metric, all carriers are being treated

consistently, which is a conceptual realization of “fairness”. In a perfectly fair situation, the value of MAD would be 0, which means all the carriers receive the equal amount of benefit, by switching from FCFS to 2OptSwap.

One way of using MAD is incorporating it into the objective function with a weight coefficient. The objective function proposed as Eq.(3) in Chapter 3 then becomes Eq.(18). In this way, minimizing the objective function becomes minimizing the summation of the total CTA delay cost, the total weighted ETA delay cost, and the weighted unfairness represented by MAD value.

$$min\ \sum_{j\in S}\sum_{i\in F}\left(c_i\ x_{ij}\ d_{ij}\right) + w^{ETA}\sum_{j\in S}\sum_{i\in F}\left(c_i\ x_{ij}\ d_{ij}^{ETA}\right) + w^{MAD}MAD \tag{18}$$

Additionally, the benefits deviation in the fairness metric could be used as a real-time metric in the rolling horizon scheme to help balance the fairness among different carriers. The idea is using the performance from previous epochs to assist decision-making in the current and future epochs. The basics of this approach are explained below.

For each carrier i , we calculate its benefits deviation dev_i: $dev_i = b_i - \overline{b}$. A positive benefits deviation means the carrier gains more benefits from switching from FCFS to 2OptSwap, than the mean benefits value among all the carrier groups. A negative benefits deviation means the carrier gains less benefits from switching from FCFS to 2OptSwap, than the mean benefits value among all the carrier groups. The absolute value of the deviation represents how much the benefits deviate from the mean value.

Given this setting, the decision making can be adjusted in multiple ways. For example, one way of adjusting it is, for those carriers with $dev_i < 0$, we increase each flight's priority label to an extent that positively correlates with $|dev_i|$. This increases the flight's probability of being swapped forward and potentially helps the carrier receive more benefits in the next period. On the contrary, for those carriers with $dev_i > 0$, we decrease each flight′s priority label to an extent that positively correlates with $|dev_i|$. This decreases the flight's probability of being swapped forward and potentially makes the carrier receive less benefits in the next period. For carriers with $dev_i = 0$, no adjustment is needed. This priority adjustment process can be formulated as Eq.(19).

$$p\_adj_j^i = p_j^i - \alpha \times dev_i \tag{19}$$

where p_j^i is the priority label for flight j in carrier i, $p\_adj_j^i$ is the adjusted priority label for flight j in carrier i, α is the adjustment rate and can be tuned in specific settings such that this adjustment works in reasonable iterations yet not drastically compromise the overall results, and dev_i is the benefits deviation for carrier i.

For our 2OptSwap scheme, the adjustment in the priority label values can only make a difference on the CTA assignment results if the swapping between two flights also passes the CTA delay cost check. We thus also make some adjustments on this rule to help facilitate this idea. For those carriers with $dev_i < 0$, we increase each flight's unit cost for CTA delay to an extent that positively correlates with $|dev_i|$. This increases the flight's probability of being swapped forward and potentially helps the

carrier receive more benefits in the next period. On the contrary, for those carriers with $dev_i > 0$, we decrease each flight's cost to an extent that positively correlates with $|dev_i|$. This decreases the flight's probability of being swapped forward and potentially makes the carrier receive less benefits in the next period. For carriers with $dev_i = 0$, no adjustment is needed. This priority adjustment process can be formulated as Eq.(20).

$$cost\_adj_j^i = cost_j^i - \beta \times dev_i \quad (20)$$

where $cost\_adj_j^i$ is the adjusted CTA delay cost per minute for flight j in carrier i, $cost_j^i$ is the CTA delay cost per minute for flight j in carrier i, β is the adjustment rate and can be tuned in specific settings such that this adjustment works in reasonable iterations yet not drastically compromise the overall results, and dev_i is the benefits deviation for carrier i. Note that this adjusted unit cost for CTA delay is only used in 2OptSwap swapping checks when the real time metric is used to adjust carrier benefits fairness. The total CTA delay cost in the performance metric is still calculated by using the original unit CTA delay cost for the flights.

Additional ideas that could help balance the fairness of CTA assignment schemes among carriers are provided as below:

- Add a compensation scheme at the end of the day. This is the most straightforward solution. The discussion and examples were already given in Section 5.6. At the end of the day, we match the average delay cost reduction for all carriers to be the same value as the average delay cost reduction for all

flights among all carriers. The advantage of this solution is that it does not require any change to the CTA assignment schemes proposed, since the compensation scheme is added independently at the end. Another advantage is that this solution guarantees all carriers gain the maximum benefits available, after the compensation. The drawback of this solution is it requires detailed carriers' agreement on the price of the delay to complete the compensation.

- Develop an auction-based CTA assignment scheme. Carriers could bid for desired CTA slots. Similar auction ideas have been used in existing studies on airspace resource allocation problems [60, 69, 70]. The drawback of this solution is that the CTA assignment problem is a real-time problem and requires real-time data, and an auction scheme could increase the real-time decision-making work from different parties.
- Create a carriers' point system. Each carrier starts with a certain number of points. Every time their flights gets swapped forward, a certain number of points will be deducted from the carrier, and vice versa, every time their flights get swapped back, a certain number of points will be added to the carrier. The starting value of the points and the adjustment amount should be carefully tested such that it does not allow gaming opportunities yet still offers carriers enough flexibility on achieving the decisions that meet their own preferences.

6.5 Chapter Conclusion

The rolling horizon approach for the simulation scheme is designed in this chapter. The CTA assignment and landing service can be implemented dynamically using this approach to incorporate the newest information, no matter which CTA assignment scheme is chosen. A fairness metric was also designed, and several ideas were proposed on how to use it to improve our system. The fairness metric can be added as a term in the objective function. It can also be used as a real-time metric to adjust future decisions in favor of a fairer benefits distribution among carriers. These ideas are further tested and implemented in Chapter 7.

Chapter 7: Dynamic Case Studies

In this chapter, simulation results are presented under the rolling horizon dynamic scheme proposed in Chapter 6.

7.1 Simulations Evaluating the Parameters in the Rolling Horizon Approach

In this section, several parameters in the rolling horizon approach are tested and the differences in the total CTA delay cost are observed.

Simulations are conducted under different look-ahead windows and rescheduling intervals. The testing AAR chosen is 90 flights/hour. Each setting is implemented in one realization.

The CTA delay cost in each scenario is recorded in Table 22 and Table 23. The columns represent different look-ahead windows, and the rows represent different rescheduling intervals. The nature of the rolling horizon approach guarantees that the look-ahead window must be equal or greater than the rescheduling interval, and that is why we only conduct experiments in the upper triangular of the tables. The cell with the look-ahead window and rescheduling interval both being 86400 s means that the whole day simulation is conducted in one interval (equivalent to using a static

system). Under each column, there is a sub column indicating the FCFS solution and a sub column indicating the 2OptSwap solution.

It was observed that the FCFS solutions are different in the two tables, even under the same parameter settings. This is because solutions in the previous epochs affect the current and future epoch solutions. In Table 22, any previous epochs are first conducted following FCFS, then finalized in 2OptSwap, while in Table 27, any previous epochs are first conducted following FCFS, then finalized in IP. These differences from previous epochs affect the current and future epoch dataset and the overall solution for all assignment schemes.

We should also keep in mind that both the 2OptSwap and the IP model use the maximum delay constraint that takes input of the CTA assigned in the FCFS scheme. As mentioned earlier, under a rolling horizon scheme, the FCFS solution is affected by the assignment happened in the previous epochs, so is the maximum delay constraint used in 2OptSwap and IP schemes.

In terms of comparison between the 2OptSwap and IP solution, we observe the same trend as in the static cases, the total CTA delay cost in 2OptSwap is higher than in the IP model, given the same parameter settings. In this set of simulations, part of this gap also results from the carrier preference/priority constraint in the 2OptSwap scheme.

Table 22 The Total CTA Delay Cost Using 2OptSwap in Rolling Horizon

look-ahead window (s)	600	1800	3600	86400

rescheduling interval (s)	FCFS	2OptSwap	FCFS	2OptSwap	FCFS	2OptSwap	FCFS	2OptSwap
600	409797	179195	400151	185501	400151	185501	400151	185501
1800			361371	229003	358816	229857	358816	229857
3600					348795	231687	347479	232803
86400							334420	237151

Table 23 The Total CTA Delay Cost Using IP in Rolling Horizon

look-ahead window (s)	600		1800		3600		86400	
rescheduling interval (s)	**FCFS**	**IP**	**FCFS**	**IP**	**FCFS**	**IP**	**FCFS**	**IP**
600	411345	166648	404634	179173	404634	179173	404634	179173
1800			360007	217034	359719	217131	359719	217131
3600					349414	220463	349123	220562
86400							334420	226326

To further evaluate the effect of the maximum delay constraint and validate that our rolling horizon approach is performing as expected, another set of simulations is conducted while excluding the maximum delay constraint. Other parameters and settings are controlled to be exactly the same as in the previous case.

The results are shown in Table 24 and Table 25. Besides the previous observations (previous epochs affect the current and future epoch dataset and the overall solution for all assignment schemes, including FCFS), it is also observed that for the IP model, solving the whole day in one interval provides the lower bound for all interval combinations. However, in real world, the system will never have the whole day information at hand ahead of time. The best we could do is to make the best decision given known information in each epoch. The good news is, in our case here, as long as we set up a large enough look-ahead time window (compared to the rescheduling

interval), we are able to achieve the same optimal solution (highlighted in grey in Table 25) as in the whole-day case (pretending the whole day information is known ahead of time). These results validate our rolling horizon approach is performing as expected.

For case studies in the succeeding sections, the maximum delay constraint is added back into the model and a look-ahead window of 1 hour and rescheduling interval of 30 minutes are selected. This gives us a reasonable and practical interval length for CTA assignment and allows us to factor in the newest information and upcoming flights in the system.

Table 24 The Total CTA Delay Cost Using 2OptSwap in Rolling Horizon (w/o MAXDLY)

look-ahead window (s)	600		1800		3600		86400	
rescheduling interval (s)	FCFS	2OptSwap	FCFS	2OptSwap	FCFS	2OptSwap	FCFS	2OptSwap
600	412674	156856	422674	154932	422674	154932	422674	154932
1800			362538	211049	363383	210112	363383	210112
3600					349889	213909	349091	214213
86400							334420	219574

Table 25 The Total CTA Delay Cost Using IP in Rolling Horizon (w/o MAXDLY)

look-ahead window (s)	600		1800		3600		86400	
rescheduling interval (s)	FCFS	IP	FCFS	IP	FCFS	IP	FCFS	IP
600	408145	156619	403893	152394	403893	152394	403893	152394
1800			376858	153445	378165	152394	378165	152394
3600					366157	153140	366940	152394
86400							334420	152394

7.2 Simulations Exploring Dynamic AAR

As stated in Section 6.1, one of the biggest advantages of implementing simulations in a dynamic setting is that the system can vary parameters such as AARs and noises throughout the day. In this section, simulations exploring dynamic AAR settings are conducted.

a) Dynamic AAR in Both CTA Assignment and Landing Service

In this set of simulations, we allow the AAR used in the CTA assignment to vary from the AAR used in the landing service, and both AARs can change throughout the day. This is a more realistic representation of the real-world situations than using the same value static AAR for the two processes. When assigning CTAs, the system uses a call rate, which is essentially a prediction of the future AAR value for when the flights arrive at the destination and could be different from the true value due to prediction error and environmental changes. The look-ahead window was set at 1 hour and the rescheduling interval was set at 30 minutes in the rolling horizon approach settings.

The same incoming flights data are used as in Chapter 4. The ATA error model used is Gaussian noise among all flights with mean = 0 s and standard deviation = 15 s. The input AAR values tested are listed in Table 26. The first column represents the hour of the day. The second column, AAR_Sched, represents the AAR values used in CTA assignment/scheduling and it can change throughout the day if a dynamic CTA assignment scheme is used. The third column, AAR_Serv, represents the AAR values

used in the landing service. It reflects the true capacity at the airport and always changes throughout the day in real conditions. It can be seen that the two AAR values vary throughout the day and in some hours, they are different from each other. For example, from Hour 0 to 6 (a left-closed and right-open interval), the two AAR values are the same, from Hour 6 to 9, the AAR_Sched is greater than AAR_Serv, from Hour 9 to 11, the AAR_Sched is smaller than AAR_Serv.

Table 26 Dynamic AAR Inputs

Hour	AAR_Sched (flights/hr)	AAR_Serv (flights/hr)
0	108	108
1	108	108
2	108	108
3	108	108
4	108	108
5	108	108
6	116	95
7	118	95
8	108	95
9	96	100
10	96	100
11	96	95
12	96	90
13	96	96
14	96	105
15	105	108
16	108	108
17	108	108
18	108	108
19	108	108
20	108	108
21	108	108
22	108	120
23	108	110

A single-realization simulation is conducted, and the result is plotted in Figure 38. The x-axis represents the time of the day. In the top subplots, the y-axis represents the

flights count per 15 minutes, and each light pink bar represents the demand of incoming flights for a 15-minute interval. The green curve is the CTA assignment rate. The dashed red line represents AAR_Sched and serves as the upper bound for the green curve, and the dotted blue line represents AAR_Serv. The variations of the AAR values from the input table are clearly shown in these lines. In the bottom subplot, the left y-axis is the CTA schedule delay for the green curve, and the right y-axis is the service delay for the brown curve. We use the delay in time instead of the delay cost in dollars because the former is directly and immediately affected by the AAR changes in the simulation settings.

As shown in Figure 38, the CTA schedule delay and the service delay in the plot change as expected throughout the day, based on different values given in the input AARs and the demand. For example, from Hour 0 to 6, the two AAR values are the same and the demand is very low, we observe minimal CTA schedule delay and service delay; from Hour 6 to 9, AAR_Sched is greater than AAR_Serv, which means we are assigning flights CTAs at a higher rate than the true rate the airport could handle by the time these flights arrive, and flights have to wait to land and be served, this incurs high service delay (brown curve in the bottom subplot); from Hour 9 to 11, the AAR_Sched is smaller than AAR_Serv. This frees up the queue at the airport for flights to land and be served, so the service delay dropped immediately at this time. We also see that when the demand is higher than the capacity, the CTA schedule delay (green curve in the bottom subplot) goes up drastically.

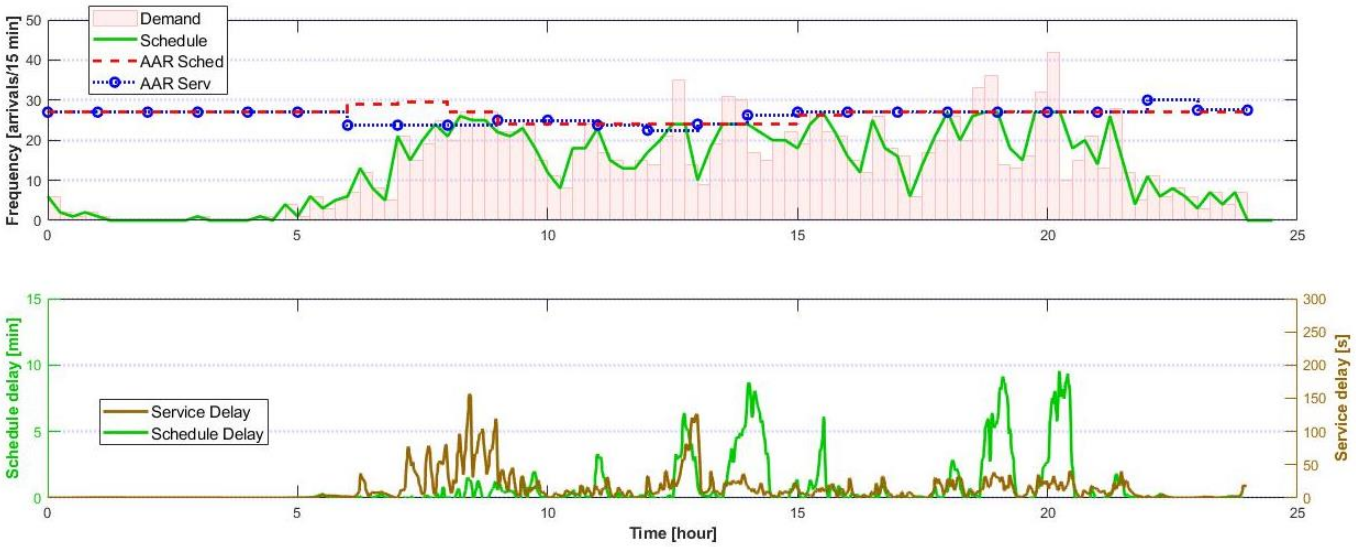

Figure 38 Simulation Using Dynamic AAR

b) Large Static AAR in CTA Assignment and Dynamic AAR in Landing Service

As a comparison, another simulation is conducted using a large static AAR_Sched value but a dynamic AAR_Serv. This corresponds to a situation where a large, static AAR is used in CTA assignment scheme, but the true capacity is changing throughout the day.

The same incoming flights data are used as in Chapter 4. The ATA error model used is Gaussian noise among all flights with mean = 0 s and standard deviation = 15 s. The AAR input table is shown in Table 27.

Table 27 Large Static AAR_Sched and Dynamic AAR_Serv Inputs

Hour	AAR_Sched (flights/hr)	AAR_Serv (flights/hr)
0	108	108

1	108	108
2	108	108
3	108	108
4	108	108
5	108	108
6	108	95
7	108	95
8	108	95
9	108	100
10	108	100
11	108	95
12	108	90
13	108	96
14	108	105
15	108	108
16	108	108
17	108	108
18	108	108
19	108	108
20	108	108
21	108	108
22	108	120
23	108	110

A single-realization simulation is conducted, and the result is plotted in Figure 39. It can be seen that the dashed red line stays flat, which means a static AAR_Sched value is used throughout the day. As expected, this setting incurs larger service delay in general, especially when AAR_Serv is smaller than AAR_Sched.

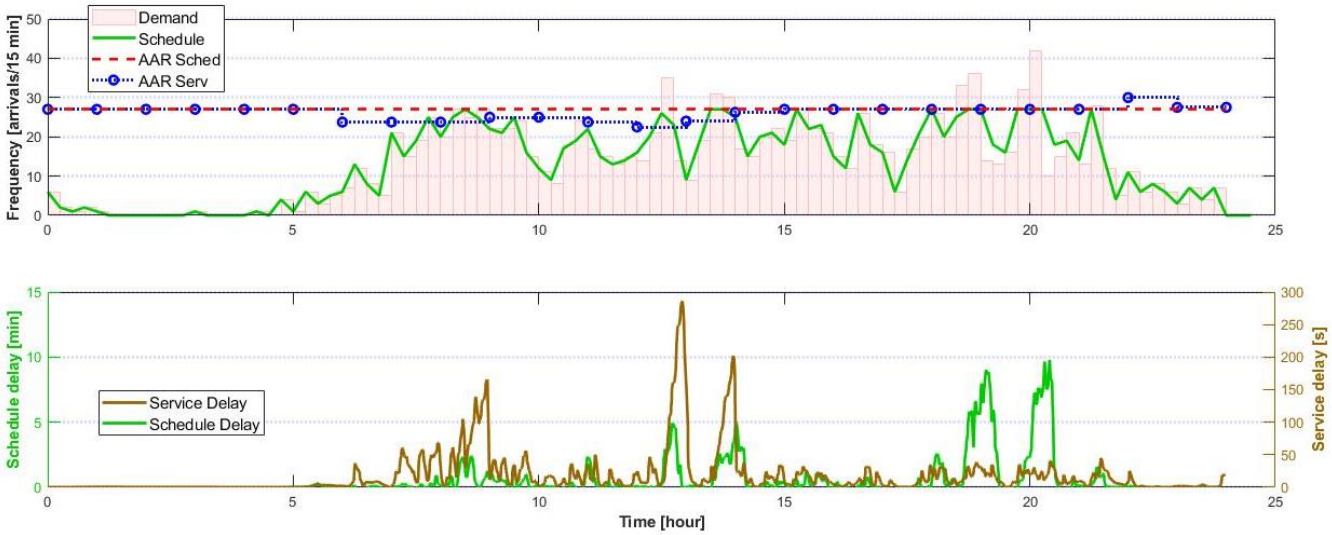

Figure 39 Simulation Using Large Static *AAR_Sched* and Dynamic *AAR_Serv*

c) Small Static AAR in CTA Assignment and Dynamic AAR in Landing Service

As a comparison, another simulation is conducted using a small static AAR_Sched value but a dynamic AAR_Serv. This corresponds to a situation where a small, static AAR is used in CTA assignment scheme, but the true capacity is changing throughout the day.

The same incoming flights data are used as in Chapter 4. The ATA error model used is Gaussian noise among all flights with mean = 0 s and standard deviation = 15 s. The AAR input table is shown in Table 28.

Table 28 Small Static *AAR_Sched* and Dynamic *AAR_Serv* Inputs

Hour	*AAR_Sched* (flights/hr)	*AAR_Serv* (flights/hr)
0	100	108

1	100	108
2	100	108
3	100	108
4	100	108
5	100	108
6	100	95
7	100	95
8	100	95
9	100	100
10	100	100
11	100	95
12	100	90
13	100	96
14	100	105
15	100	108
16	100	108
17	100	108
18	100	108
19	100	108
20	100	108
21	100	108
22	100	120
23	100	110

A single-realization simulation is conducted, and the result is plotted in Figure 40. It can be seen that the dashed red line stays flat, which means a static AAR_Sched value is used throughout the day. As expected, this setting incurs larger CTA schedule delay since a relatively small AAR_Sched is used, compared to previous cases.

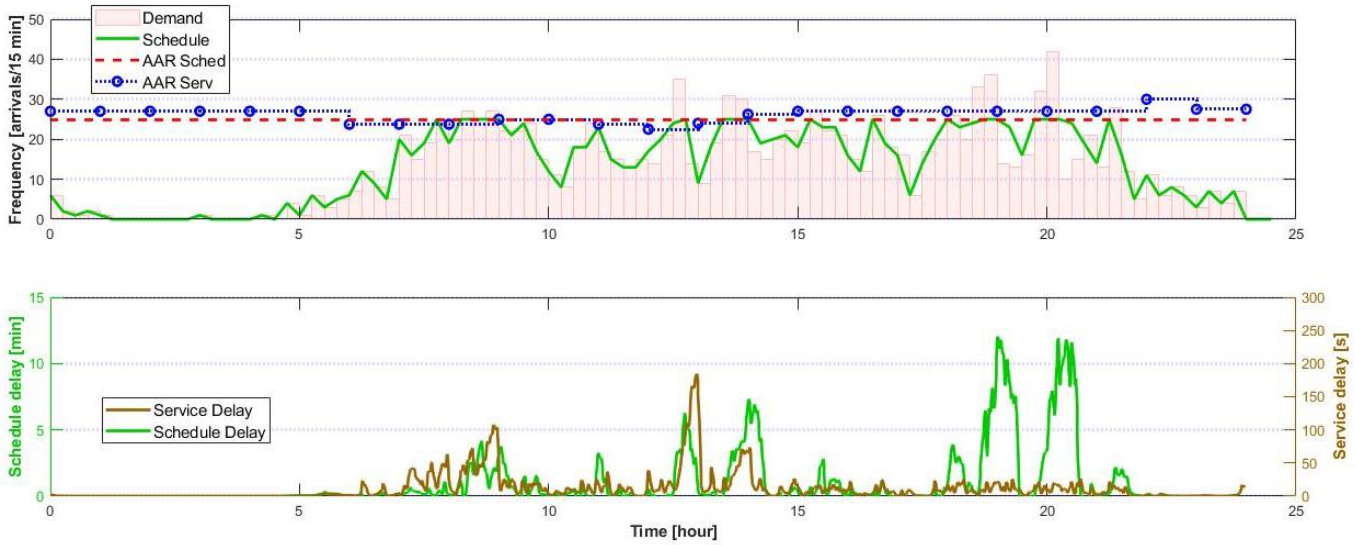

Figure 40 Simulation Using Small Static *AAR_Sched* and Dynamic *AAR_Serv*

The above three simulations indicates that in real-world applications, it is important to use the up-to-date information and use a close-to-truth value for the AAR in CTA assignment schemes. This could help avoid unnecessary delays in the systems. Our simulation scheme can handle such a dynamic input throughout the day and perform as expected.

7.3 Simulations Exploring Dynamic AAR and Noise

In this section, dynamic noise control is incorporated into the system, in addition to the dynamic AAR control.

Dynamic noise refers to the noise used in the ATA error model being dynamic. Under such a setting, flights meet their CTAs following different noise distributions

throughout the day. This is more a realistic representation of the real world than using a static noise distribution throughout the day, since the factors that affect flights meeting their CTAs are constantly changing.

The same incoming flights data are used as in Chapter 4 and the same dynamic AAR values are used as Table 26 in Section 7.2 a). The ATA error model is varied throughout the day by varying the standard deviation σ in the Gaussian distribution (the mean of this distribution stays at a fixed value, 0 s). These input values are listed in Table 29.

Table 29 Dynamic AAR and Noise Inputs

Hour	AAR_Sched (flights/hr)	AAR_Serv (flights/hr)	σ of Noise (s)
0	108	108	10
1	108	108	10
2	108	108	10
3	108	108	10
4	108	108	10
5	108	108	10
6	116	95	10
7	118	95	10
8	108	95	10
9	96	100	0
10	96	100	0
11	96	95	0
12	96	90	0
13	96	96	0
14	96	105	0
15	105	108	10
16	108	108	10
17	108	108	10
18	108	108	10
19	108	108	20
20	108	108	20
21	108	108	20

22	108	120	20
23	108	110	20

A single-realization simulation is conducted, and the result is plotted in Figure 41. It can be seen that the CTA schedule delay and the schedule delay in the plot changes as expected throughout the day, based on different values of the AAR, noise, and the demand. In addition to the observations in Case a) in Section 7.2, we observed how the changes in noise affect the service delay. For example, from Hour 9 to 15, the noise in the ATA error model reduced to 0 s, so the service delay curve (brown curve in the bottom subplot) stayed near 0 s except for when the AAR_Sched is greater than AAR_Serv (Hour 11 and 12). During Hour 15 to 19, the noise increased compared to before Hour 15, so the service delay increased. From Hour 19 and after, the noise increased again compared to before Hour 19, so the service delay increased to a higher level.

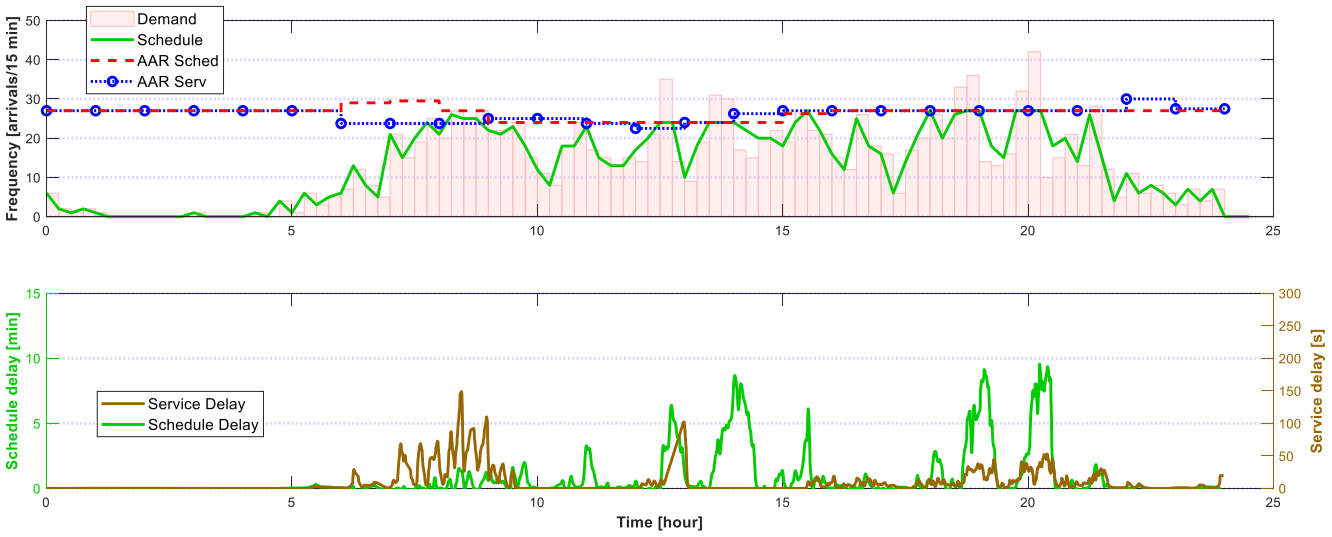

Figure 41 Simulation Using Dynamic AAR and Noise

7.4 Simulations Exploring the Real Time Fairness Metric and Fairness Adjustment Methods

In this section, two sets of simulation experiments are provided to evaluate the effects of using the real time fairness metric in a dynamic setting, following the ideas proposed in Section 6.4.

The first set of simulations serves as a control group: we use the rolling horizon approach, but we do not use any real-time metrics or any adjustments. The second set of simulations serve as the experimental group: we use the rolling horizon approach and use the real time fairness metric to adjust decisions in future epochs.

We tested 7 AAR values in [80, 140] and 3 ETA delay scenarios (low, medium, high), and for each of the 21 combination settings, we repeated 500 Monte Carlo realizations in the simulation. The mean values of the switching benefits (defined as the average CTA delay cost reduction per flight from FCFS to 2OptSwap in Section 6.4) for different carriers were plotted in the boxplots in Figure 42 and Figure 43. Figure 42 shows the metric for all 36 carriers, and Figure 43 only shows the metric for the carriers that have more than 5 flights of the day (referred to as "large carriers"). The real carrier names are anonymized as Carrier A, Carrier B, …, for sensitivity reasons. For each box, the red bar represents the median (out of the 21 data points) in that carrier. It can be seen that the average CTA delay reduction is quite uneven among carriers. Several carriers even receive negative benefits when switching from FCFS to 2OptSwap. A comparison between the two figures tells us

that carriers who have a small number of flights tend to have more extreme values on switching benefits.

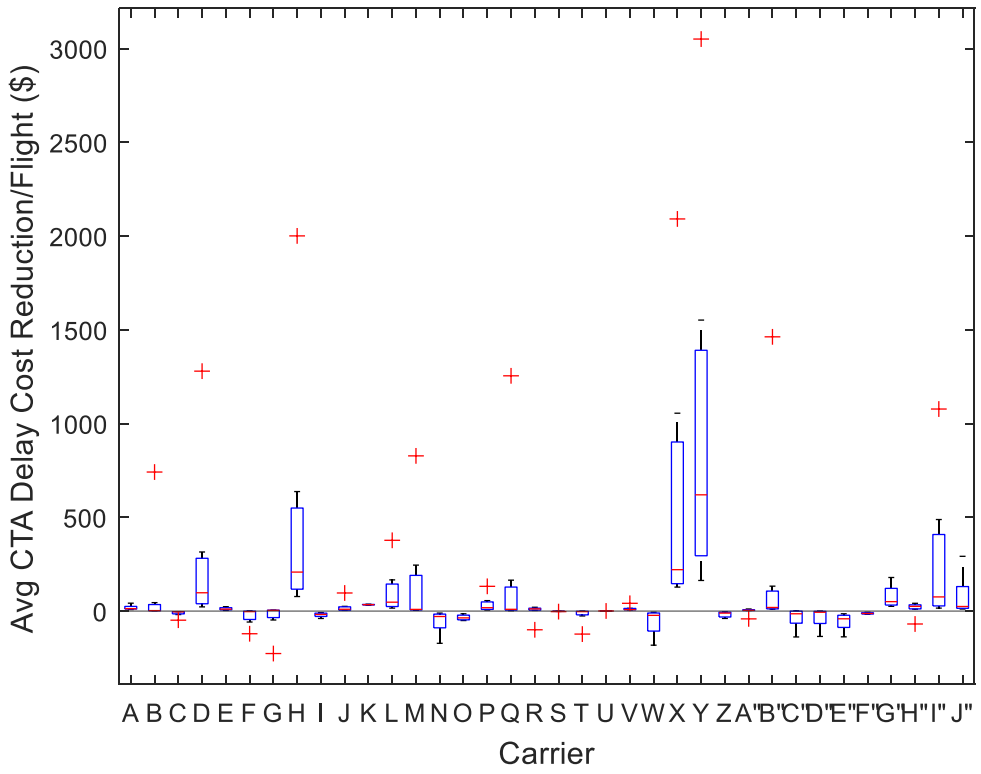

Figure 42 The Average CTA Delay Cost Reduction among All Carriers (w/o Adjustments)

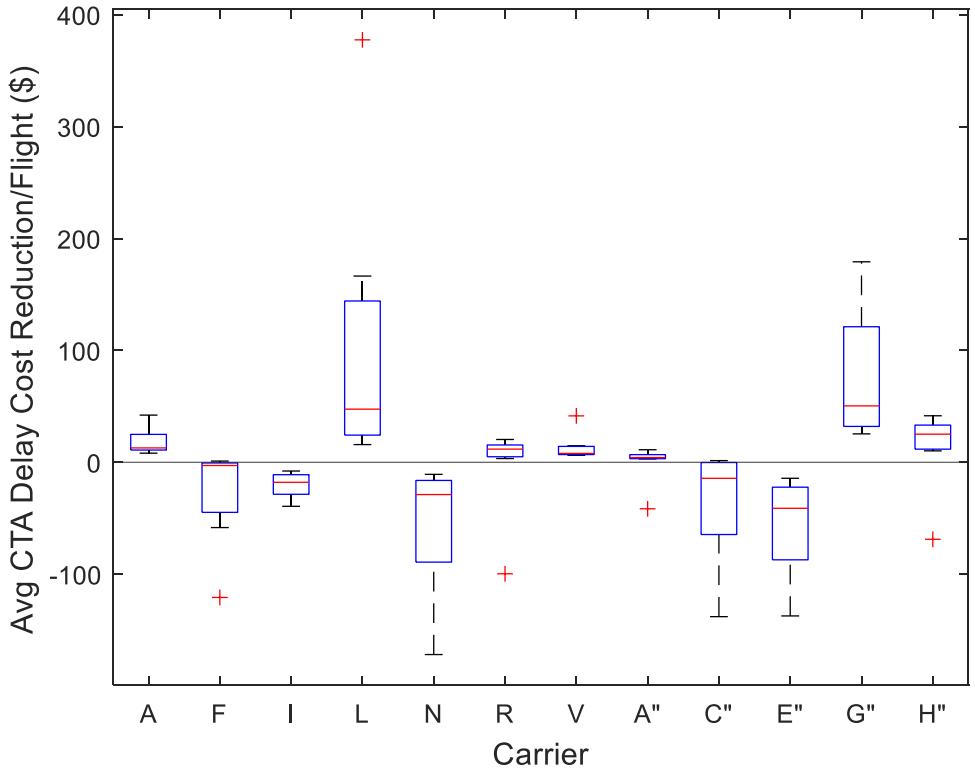

Figure 43 The Average CTA Delay Cost Reduction among Large Carriers (w/o Adjustments)

The experimental group simulations are then conducted. We used the same 7 AAR values in [80, 140] and 3 ETA delay scenarios (low, medium, high) as in the control group, and the process is repeated in 500 Monte Carlo realizations. The only difference between the control group and the experimental group is in the latter, we add the real time fairness metric and use it to adjust CTA assignments in future epochs following the methods proposed in Section 6.4. The values of the adjustment rates used are as follows:

- $\alpha, \beta = 2$ for those carriers that have $b_i < 0$.

- $\alpha, \beta = 1$ for all other carriers' flights.

The reason for using a higher adjustment rate for those carriers that have negative benefits in previous assignments is that we want to help avoid any carrier receiving negative benefit at the end of the day. A higher adjustment rate allows faster adjustments.

The results are shown in Figure 44 and Figure 45. Figure 44 shows the metric for all 36 carriers, and Figure 45 only shows the metric for the carriers that have more than 5 flights during the day (referred to as "large carriers"). Comparing to Figure 42 and Figure 43 in the control group, we notice that in the experiment group, the variance of the switching benefits among carriers are smaller. In Figure 45 (for large carriers), most of the switching benefits are above zero. There are five carriers with median switching benefits below zero in Figure 43: F, I, N, C'', E'', and their values are all adjusted to above or near zero in Figure 45.

These results indicate that by using our proposed real time metric and fairness adjustment methods, the system can achieve a fairer switching benefits distribution among carriers and help avoid negative benefits for any carrier.

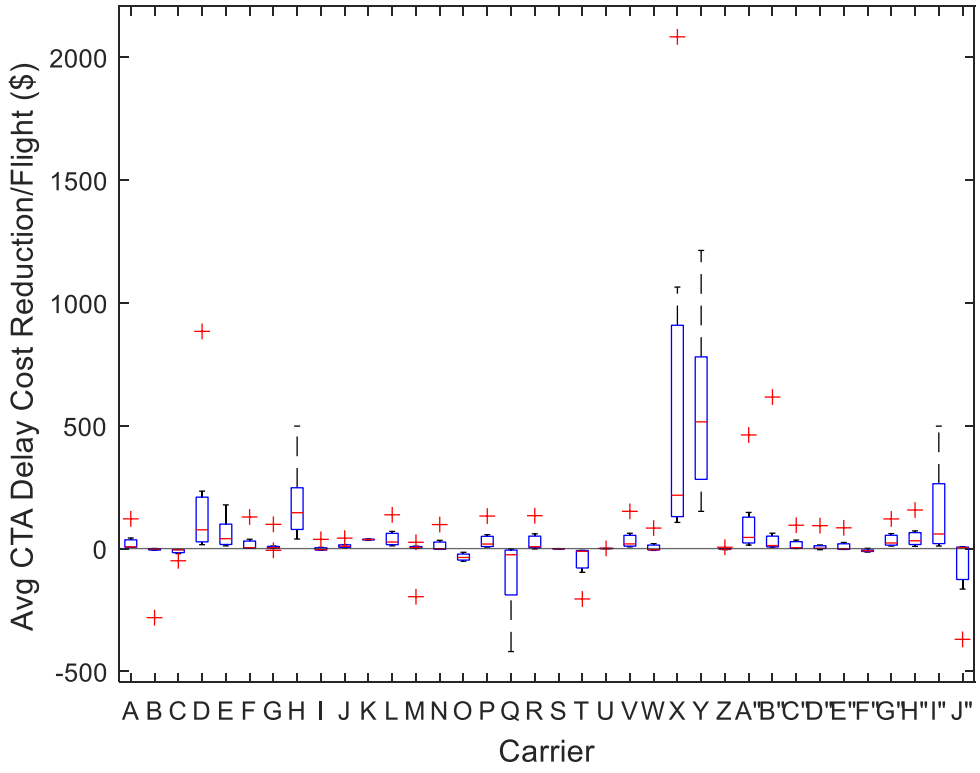

Figure 44 The Average CTA Delay Cost Reduction among All Carriers (w. Adjustments)

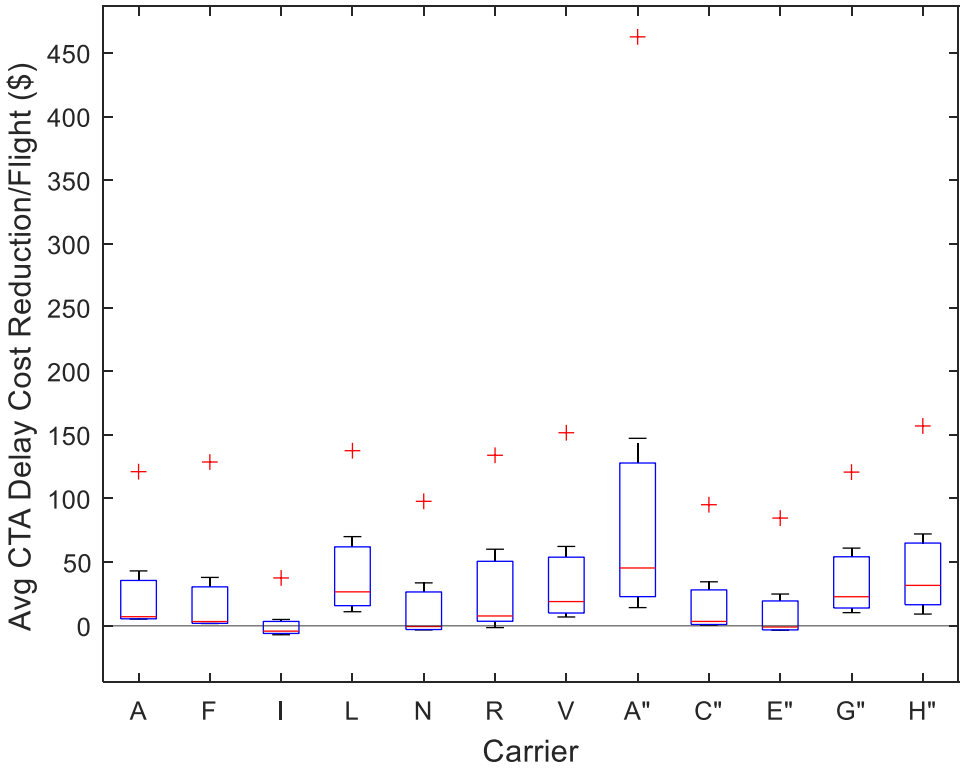

Figure 45 The Average CTA Delay Cost Reduction among Large Carriers (w. Adjustments)

The MAD metric from the two experiments and their differences are summarized in Table 30. The second column, $\bar{b}\_w/o\_adj$, is the average switching benefits per flight over all carriers when no fairness adjustments are made. The third column, $\bar{b}\_w\_adj$, is the average switching benefits per flight over all carriers when fairness adjustments are made. The fourth column, MAD_w/o_adj, is the MAD for the simulations where no fairness adjustments were made. The fifth column, MAD_w_adj, is the MAD for the simulations where the real time fairness adjustments were made. The sixth column is the reduction of MAD from not using adjustments to using adjustments, and the seventh column is the reduction percentage of it.

It can be seen that by using the proposed fairness adjustments, the MAD was reduced 8% to 60%, under different AAR values. Recall the MAD definition in Section 6.4; a large MAD value is interpreted as unfair, and a small MAD value is interpreted as fair. This means that the switching benefits distribution while using the fairness adjustments is fairer compared to not using them.

It is worth mentioning that by comparing $\bar{b}\_w\_adj$ with $\bar{b}\_w/o\_adj$, we notice that adding the fairness adjustments makes the overall solution worse in terms of the average CTA delay cost reduction. This is not surprising since during the adjustments, the system uses the adjusted cost value, rather than the true cost value as a swapping criterion. Adding the adjustments could allow swaps that increase the overall total CTA delay cost, as a tradeoff for a fairer benefits distribution.

We should keep in mind that no matter how good the real time metric works, it will never have the opportunity to adjust fairness for those carriers that have only 1 flight, because it can adjust flights only in the future epochs. Similarly, the method has very limited ability to adjust fairness for those carriers that have only a few flights. This explains why extreme values still exist after adjustments for those small carriers that have few flights as shown in Figure 44.

Table 30 The Average Benefits Values and MAD Values

AAR	$\bar{b}\_w/o\_adj$	$\bar{b}\_w\_adj$	MAD_w/o_adj	MAD_w_adj	$MAD\_Reduction$	$MAD\_Reduction\%$
80	370.0	157.2	581	230	351	60%
90	127.3	98.3	203	161	42	21%
100	59.0	57.5	95	85	10	10%
110	37.4	37.0	59	54	5	8%
120	26.0	24.7	40	36	4	9%

130	18.4	17.0	28	25	3	10%
140	13.9	12.4	21	18	3	14%

Higher adjustment rates (α and β) are also tested in additional simulations. Higher adjustment rates help achieve a better solution in terms of fairness, but also tend to cause the CTA delay cost in 2OptSwap to be higher (due to using the adjusted cost value, rather than the true cost value in swapping criteria). The fairness in carriers' benefits distribution and the overall CTA delay cost reduction is always a trade-off and we cannot just sacrifice one for the other.

Overall, we conclude that the usage of a real time fairness metric for adjustments helps the system achieve a fairer benefits distribution but does not guarantee perfect fairness, and it can be combined with a post-schedule benefits match up to achieve a perfectly fair distribution where all the carriers could get the same amount of benefits (As proposed in Section 5.6). Day-to-day adjustments could also be added such that the benefits distribution in previous days could be counted in decision-making for future days.

7.5 Chapter Conclusion

This chapter implemented the rolling horizon approach, the fairness metric, and the fairness adjustment methods proposed in Chapter 6.

The simulation experiments in Section 7.1 showed that the look-ahead time window and the rescheduling interval work as expected in the rolling horizon design.

In Section 7.2 and 7.3, simulation results proved that our system could handle dynamic AAR values and noise values and provide solutions using the most up-to-date information.

In Section 7.4, we implemented the fairness metric, and the fairness adjustment methods. Results show that these methods can help relieve the unfairness in benefits distribution among carriers. However, there is always a trade-off between minimizing the total CTA delay cost and minimizing the total benefits deviation among carriers. A post-schedule benefits match up can still be used in addition to the fairness adjustment method for a perfect fair benefits distribution.

Chapter 8: Conclusions and Future Research

8.1 Conclusions

In this dissertation, several CTA assignment schemes were proposed for incorporating CDM concepts into a flight arrival management system such as TBFM. The proposed schemes take user preferences into consideration while assigning flights CTAs, aiming at reducing the total CTA delay cost. The CTA assignment scheme under FCFS was used as a baseline model to show the efficiency and benefits of the proposed schemes.

Monte Carlo simulations for these schemes were designed and real historical flight data with noise variations were used in the simulations. Detailed approaches on data preparation and simulation steps were provided. The approaches on simulating the flights priority can be replaced by the actual priority list provided by the carriers when being applied to the real world. Metrics such as the total CTA delay cost, the total service delay cost, and the runtimes are tracked to evaluate the performance among different schemes in various parameter settings. These steps can be adopted to allow study of any particular airport situation.

Simulation results show that our proposed heuristic – 2OptSwap can achieve 20% to 30% reduction in total CTA delay cost compared to the FCFS baseline model and stay within 1% to 10% gap from the optimal solution with runtime 7 to 40 times

faster than the optimization model. The 2OptSwap approach can achieve similar levels of reduction from the baseline model regardless of the delay status of the incoming flights. By extending the decision-making points, the approach could achieve even more delay cost reductions. Results also show that there is an uneven level of performance distribution among different carriers.

A dynamic system was then designed using the rolling horizon approach. Under this setting, the system could take dynamic inputs such as dynamic AARs and dynamic noise distributions. We also defined a fairness metric and the corresponding fairness adjustment methods to help resolve the unfairness of benefits distribution among carriers. Results show that our proposed methods can achieve a fairer benefits distribution and the fairness metric can improve 8% to 60%, under different AAR values.

8.2 Future Research

Future research could be conducted in several directions:

A significant contribution of this project is the refinement of the simulation platform to be able to handle a wider variety of CTA assignment schemes. In this vein, other schemes could be developed and tested. For example, the goals could be changed from minimizing the total CTA delay cost to other metrics, or a combination of several metrics.

More comprehensive cost functions could be applied. For example, fuel consumption cost could be added to the simulations, if a comprehensive aircraft performance model is invoked. Cost terms for cargo aircraft could also be included.

As discussed in Section 5.6, Section 6.4, and Section 7.4, additional ideas could be implemented to achieve more evenly distributed performance among different carriers.

The inter-carrier swapping idea could be adjusted to adapt different versions of carrier-expressed preferences.

The challenges of obtaining airline cost/preference information could be further addressed, while ensuring that the airlines are encouraged to provide truthful information and ensuring airlines are treated fairly. The proposed ranking-based cost system could also be implemented for when flight unit delay cost is missing, as proposed in Section 4.2.3.

As mentioned in Section 3.2.2, additional constraints could be added to better simulate the real-world situations. Examples would include “non-changeable” constraints on some flights to simulate the situation where flights within a given distance on the same air route cannot be swapped, and “non-compliance” constraints for some flights to represent that some flights are assigned CTAs, but will not adjust operations to meet the CTAs. Flights with EDCTs could also be treated differently to avoid double delays.

This study only conducts CTA assignment at one freeze horizon boundary. In the future, multiple decision points can be combined, such as the Coupled Metering Point (CMP) freeze horizon and the Extended Metering Point (XMP) freeze horizon. An advantage to this kind of scheme is that a more robust re-organization of the CTA assignment can be accomplished over a greater distance, with the understanding that the CTA order need not stay the same between these different metering points. Thus, more real-time information on flight progress could be accommodated, as could a multi-stage en route CTA assignment strategy. This idea also aligns with the "extended metering" concept discussed in Section 5.7.

The CTA assignment problem could be developed in an integrated problem combining ALP, with detailed runway assignments. However, such an integrated problem brings challenges on selecting the correct AAR value to use, since the solution from the ALP could affect AAR, and its upstream process, the CTA assignment problem, needs AAR as an input.

One airport could have multiple metering fixes and flights from different directions should use different metering fixes. More detailed simulations allowing multiple metering fixes at one airport could be developed if given detailed flights trajectory data. Trajectory-based flight management and simulations could also be designed given appropriate data and tools.

References

1. Federal Aviation Administration, FAA Order JO 7210.3CC - Facility Operation and Administration. FAA National Headquarters (FOB−10B) Publications & Administration - (AJV-P12).

2. Federal Aviation Administration. Time Based Flow Management. [cited 2019 Dec.]; Available from: https://www.faa.gov/nextgen/cip/tbfm/.

3. Federal Aviation Administration. AMPM Help Glossary. Available from: https://aspm.faa.gov/aspmhelp/index/Glossary.html.

4. Federal Aviation Administration. Instrument Procedures Handbook (IPH) Glossary. 2017; Available from: https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/instrument_procedures_handbook/.

5. Wambsganss, M., Collaborative Decision Making through Dynamic Information Transfer. *Air Traffic Control Quarterly*, 1996. **4**: p. 109-125.

6. Gorman, P.E., J. Hofmann, and M. Wambsganss. Collaborative decision making between the Federal Aviation Administration and the air transport industry. in *the 16th AIAA/IEEE Digital Avionics Systems Conference*. 1997. Irvine, CA.

7. Ball, M.O., R.L. Hoffman, D. Knorr, J. Wetherly, and M. Wambsganss. Assessing the benefits of collaborative decision making in air traffic management. in *the 3rd USA/Europe Air Traffic Management R&D Seminar*. 2000. Napoli, Italy.

8. Gilbo, E., K. Howard, and A. Corp. Collaborative Optimization of Airport Arrival and Departure Traffic Flow Management Strategies for CDM. in *the 3rd USA/Europe Air Traffic Management R&D Seminar*. 2000. Napoli, Italy.

9. Chang, K., K. Howard, R. Oiesen, L. Shisler, M. Tanino, and M.C. Wambsganss, Enhancements to the FAA Ground-Delay Program Under Collaborative Decision Making. *INFORMS Journal on Applied Analytics*, 2001. **31**(1): p. 57-76.

10. Sheth, K. and S. Gutierrez-Nolasco, Incorporating User Preferences in Collaborative Traffic Flow Management, in *AIAA Guidance, Navigation and Control Conference and Exhibit*. 2008.

11. Rios, J., K. Sheth, and S. Guitierrez-Nolasco, Incorporating User Preferences within an Optimal Traffic Flow Management Framework, in *AIAA Guidance, Navigation, and Control Conference*. 2010.

12. Vakili, N., Preference Based Fair Allocation of Limited Airspace Resources. 2009, University of Maryland, College Park.

13. Vlachou, K. and D.J. Lovell, Mechanisms for Equitable Resource Allocation When Airspace Capacity is Reduced. *Transportation Research Record*, 2013. **2325**(1): p. 97-102.

14. Raj Mohanavelu Umamagesh, P., Mechanisms for Trajectory Options Allocation in Collaborative Air Traffic Flow Management. 2018, University of Maryland, College Park.

15. Idris, H., C. Chin, and A. Evans, Accrued delay application in trajectory-based operations, in *the 13th USA/Europe Air Traffic Management R&D Seminar*. 2019: Vienna, Austria.

16. Xu, Y., R. Dalmau, M. Melgosa, A. Montlaur, and X. Prats, A framework for collaborative air traffic flow management minimizing costs for airspace users: Enabling trajectory options and flexible pre-tactical delay management. *Transportation Research Part B: Methodological*, 2020. **134**: p. 229-255.

17. Carlson, P.M., Exploiting the Opportunities of Collaborative Decision Making: A Model and Efficient Solution Algorithm for Airline Use. *Transportation Science*, 2000. **34**(4): p. 381-393.

18. Ball, M.O., R. Hoffman, A.R. Odoni, and R. Rifkin, A Stochastic Integer Program with Dual Network Structure and Its Application to the Ground-Holding Problem. *Operations Research*, 2003. **51**(1): p. 167-171.

19. Sherali, H.D., R.W. Staats, and A.A. Trani, An Airspace Planning and Collaborative Decision-Making Model: Part I—Probabilistic Conflicts, Workload, and Equity Considerations. *Transportation Science*, 2003. **37**(4): p. 434-456.

20. Vossen, T.W.M. and M.O. Ball, Slot Trading Opportunities in Collaborative Ground Delay Programs. *Transportation Science*, 2006. **40**(1): p. 29-43.

21. Bencheikh, G., J. Boukachour, and A. El Hilali Alaoui, A memetic algorithm to solve the dynamic multiple runway aircraft landing problem. *Journal of King Saud University - Computer and Information Sciences*, 2016. **28**(1): p. 98-109.

22. Andreussi, A., L. Bianco, and S. Ricciardelli, A Simulation Model for Aircraft Sequencing in the Near Terminal Area. *IFAC Proceedings Volumes*, 1978. **11**(1): p. 1551-1558.

23. Andreussi, A., L. Bianco, and S. Ricciardelli, A simulation model for aircraft sequencing in the near terminal area. *European Journal of Operational Research*, 1981. **8**(4): p. 345-354.

24. Dear, R.G. and Y.S. Sherif, The dynamic scheduling of aircraft in high density terminal areas. *Microelectronics Reliability*, 1989. **29**(5): p. 743-749.

25. Balakrishnan, H. and B.G. Chandran, Algorithms for Scheduling Runway Operations Under Constrained Position Shifting. *Operations Research*, 2010. **58**(6): p. 1650-1665.

26. Malaek, S.M.B. and E. Naderi. A New Scheduling Strategy for Aircraft Landings under Dynamic Position Shifting. in *2008 IEEE Aerospace Conference*. 2008.

27. Mesgarpour, M., C.N. Potts, and J.A. Bennell. Models for aircraft landing optimization. in *The 4th International Conference on Research in Air Transportation (ICRAT 2010)*. 2010.

28. Beasley, J.E., M. Krishnamoorthy, Y.M. Sharaiha, and D. Abramson, Scheduling Aircraft Landings—The Static Case. *Transportation Science*, 2000. **34**(2): p. 180-197.

29. Balakrishnan, H. and B. Chandran, Scheduling Aircraft Landings Under Constrained Position Shifting, in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. 2006.

30. Lee, H. and H. Balakrishnan, A Study of Tradeoffs in Scheduling Terminal-Area Operations. *Proceedings of the IEEE*, 2008. **96**(12): p. 2081-2095.

31. Rodríguez-Díaz, A., B. Adenso-Díaz, and P.L. González-Torre, Improving aircraft approach operations taking into account noise and fuel consumption. *Journal of Air Transport Management*, 2019. **77**: p. 46-56.

32. Coppenbarger, R.A., R.W. Mead, and D.N. Sweet, Field Evaluation of the Tailored Arrivals Concept for Datalink-Enabled Continuous Descent Approach. *Journal of Aircraft*, 2009. **46**(4): p. 1200-1209.

33. Jones, J.C., D.J. Lovell, and M.O. Ball, Stochastic Optimization Models for Transferring Delay Along Flight Trajectories to Reduce Fuel Usage. *Transportation Science*, 2018. **52**(1): p. 134-149.

34. ATH Group. Attila™ wins ATC Maastricht 2005 Award for Innovation. Available from: https://athgrp.com/attilaaward.html.

35. Chandler, J.G., Attila Arrives. *Aviation Week & Space Technology*, 2013: p. 42-43.

36. Croes, G.A., A Method for Solving Traveling-Salesman Problems. *Operations Research*, 1958. **6**(6): p. 791-812.

37. Hao, Y., D.J. Lovell, M.O. Ball, S. Torres, and G.M. Nagle, An Arrival Scheduling Model for Incorporating Collaborative Decision-Making Concepts into Time-Based Flow Management, in *9th International Conference for Research in Air Transportation (ICRAT)*. 2020.

38. Hao, Y., S. Torres, D.J. Lovell, and M.O. Ball. Incorporating User Preferences in Time-Based Flow Management Operations. in *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*. 2020.

39. Kirkpatrick, S., C.D. Gelatt, and M.P. Vecchi, Optimization by Simulated Annealing. *Science*, 1983. **220**(4598): p. 671-680.

40. Černý, V., Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 1985. **45**(1): p. 41-51.

41. Johnson, D.S., C.R. Aragon, L.A. McGeoch, and C. Schevon, Optimization by Simulated Annealing: An Experimental Evaluation; Part I, Graph Partitioning. *Operations Research*, 1989. **37**(6): p. 865-892.

42. Torres, S. and G.M. Nagle, Analysis of Prediction Uncertainty in Interval Management, in *16th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*. 2016: Washington, D.C.

43. FlightAware. KATL Airport Flight Tracker. 2019 [cited 2019 Oct.]; Available from: https://flightaware.com/live/airport/KATL.

44. Schwartz, B., S. Benjamin, S. Green, and M. Jardin, Accuracy of RUC-1 and RUC-2 Wind and Aircraft Trajectory Forecasts by Comparison with ACARS Observations. *Weather and Forecasting*, 2000. **15**: p. 313-326.

45. Schwartz, B.E., S.G. Benjamin, S.M. Green, and M.R. Jardin, Accuracy of RUC-1 and RUC-2 Wind and Aircraft Trajectory Forecasts by Comparison with ACARS Observations. *Weather and Forecasting*, 2000. **15**(3): p. 313-326.

46. Pita, J.P., C. Barnhart, and A.P. Antunes, Integrated Flight Scheduling and Fleet Assignment Under Airport Congestion. *Transportation Science*, 2013. **47**(4): p. 477-492.

47. Ferguson, J., A.Q. Kara, K. Hoffman, and L. Sherry, Estimating domestic US airline cost of delay based on European model. *Transportation Research Part C: Emerging Technologies*, 2013. **33**: p. 311-323.

48. DVB Bank SE, An Overview of Commercial Aircraft 2018-2019. 2017: Schiphol, The Netherlands.

49. FAA. Aircraft Registry. 2019; Available from: https://www.faa.gov/licenses_certificates/aircraft_certification/aircraft_registry/releasable_aircraft_download/.

50. FlyRadius. Aircraft Information. 2017; Available from: https://www.flyradius.com/aircraft-information.

51. Controller. https://www.controller.com/. 2019; Available from: https://www.controller.com/.

52. ICAO. DOC 8643 – Aircraft Type Designators. 2020 [cited 2020 January 14]; Available from: https://www.icao.int/publications/DOC8643/Pages/Search.aspx.

53. Torres, S., Leidos independent analysis using publicly available data (unpublished). 2019.

54. FAA. Air Traffic Organization Policy, Facility Operation and Administration. 2011; Available from: http://tfmlearning.faa.gov/Publications/atpubs/FAC/1007.html.

55. Federal Aviation Administration. Aviation System Performance Metrics (ASPM). 2019; Available from: https://aspm.faa.gov/.

56. Federal Aviation Administration. ATL AAR. [cited 2019 Oct. 2019]; Available from: https://www.fly.faa.gov/Information/east/ztl/atl/atl_aar.htm.

57. Kolmogorov, A., Sulla Determinazione Empirica di una Legge di Distributione. *Giornale dell'Istituto Italiano degli Attuari*, 1933. **4**: p. 1–11.

58. Smirnov, N., Table for Estimating the Goodness of Fit of Empirical Distributions. *The Annals of Mathematical Statistics*, 1948. **19**(2): p. 279-281, 3.

59. Feller, W., On the Kolmogorov-Smirnov Limit Theorems for Empirical Distributions. *The Annals of Mathematical Statistics*, 1948. **19**(2): p. 177-189, 13.

60. Ball, M., G. Donohue, and K. Hoffman, Auctions for the Safe, Efficient and Equitable Allocation of Airspace System Resources, in *Combinatorial Auctions*. 2005, The MIT Press.

61. Isaacson, D., S. Harrison, K. Sheth, and C. Gong, Airspace Technology Demonstration – 3 (ATD-3). 2019, NASA Ames Research Center: Moffett Field, CA.

62. Swol, C.D., S. Stalnaker, and P. Coats, Simulation-Based Analysis of Early Scheduling in the Time-Based Flow Management (TBFM) System for Flights with Expect Departure Clearance Times (EDCT), in *2018 Aviation Technology, Integration, and Operations Conference*.

63. Marquant, J.F., R. Evins, and J. Carmeliet, Reducing Computation Time with a Rolling Horizon Approach Applied to a MILP Formulation of Multiple Urban Energy Hub System. *Procedia Computer Science*, 2015. **51**: p. 2137-2146.

64. Kopanos, G.M. and E.N. Pistikopoulos, Reactive Scheduling by a Multiparametric Programming Rolling Horizon Framework: A Case of a Network of Combined Heat and Power Units. *Industrial & Engineering Chemistry Research*, 2014. **53**(11): p. 4366-4386.

65. Rodrigues, M.T.M., L. Gimeno, C.A.S. Passos, and M.D. Campos, Reactive scheduling approach for multipurpose chemical batch plants. *Computers & Chemical Engineering*, 1996. **20**: p. S1215-S1220.

66. Silvente, J., G.M. Kopanos, E.N. Pistikopoulos, and A. Espuña, A rolling horizon optimization framework for the simultaneous energy supply and demand planning in microgrids. *Applied Energy*, 2015. **155**: p. 485-501.

67. Zhu, Z., X. Hu, and J. Caverlee, Fairness-Aware Tensor-Based Recommendation. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018: p. 1153 to 1162; .

68. Gupta, S., A. Jalan, G. Ranade, H. Yang, and S. Zhuang. Too Many Fairness Metrics: Is There a Solution? . 2020 [cited 2021; Available from: https://ssrn.com/abstract=3554829 or http://dx.doi.org/10.2139/ssrn.3554829.

69. Le, L., G. Donohue, and C.-H. Chen, Auction-Based Slot Allocation for Traffic Demand Management at Hartsfield Atlanta International Airport: A Case Study. *Transportation Research Record*, 2004. **1888**(1): p. 50-58.

70. Ball, M., F. Berardino, and M. Hansen, The use of auctions for allocating airport access rights. *Transportation Research Part A-Policy and Practice*, 2017. **114**: p. 186-202.