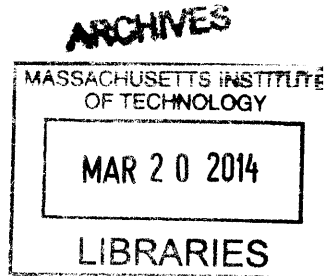


Airport Surface Traffic Optimization and Simulation in the Presence of Uncertainties

by
Hanbong Lee

S.M., Massachusetts Institute of Technology (2008)
Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Aeronautics and Astronautics
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
February 2014



© Massachusetts Institute of Technology 2014. All rights reserved.

Author
Department of Aeronautics and Astronautics
October 9, 2013

Certified by
Hamsa Balakrishnan
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Certified by
R. John Hansman
Professor of Aeronautics and Astronautics
Thesis Committee Member

Certified by
Richard Jordan
Principal Data Scientist, Verifi LLC
Thesis Committee Member

Accepted by
Paulo C. Lozano
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Airport Surface Traffic Optimization and Simulation in the Presence of Uncertainties

by

Hanbong Lee

Submitted to the Department of Aeronautics and Astronautics
on October 9, 2013, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Aeronautics and Astronautics

Abstract

Surface traffic congestion causes significant taxi delays and long queues for takeoffs at busy airports, increasing operational and environmental costs. These impacts may be mitigated by optimizing runway and taxiway schedules. In prior research, runway scheduling algorithms and taxiway schedule optimization models have been developed independently, but they are closely related in airport operations. This motivates the development of a fast and efficient algorithm for solving both scheduling problems simultaneously. While the current surface traffic optimization is mainly based on a deterministic model, there exist lots of uncertainties in airport operations. These uncertainty factors can affect airport performance, but their impacts have not been adequately understood so far.

In this thesis, two different approaches for airport surface traffic optimization are presented. The first is an integrated approach based on a mixed-integer linear programming (MILP) model to optimize both taxiway and runway schedules simultaneously, while the second is a sequential approach that combines independent runway and taxiway scheduling algorithms. The two optimization approaches are compared using various flight schedule scenarios at Detroit airport (DTW).

The second part of the thesis compares two types of control concepts for surface traffic management. The individual aircraft trajectory-based control uses the optimal solution of the surface traffic optimization as control inputs, whereas the aggregate queue-based control maintains the number of taxiing-out aircraft on the ground below a given departure queue capacity control parameter. These two control concepts are implemented in the SIMMOD environment with the same flight schedules and evaluated in terms of various airport performance metrics.

The last part of the thesis deals with the impacts of uncertainties on airport performance. Through stochastic simulations using SIMMOD, various sources of uncertainty, such as pushback times, runway exit times, taxi speeds, and runway separation times, are evaluated using flight schedules at DTW. The simulation results show that ground delays increase with an increase in uncertainty levels for most scenarios. However, the surface traffic optimization based on a deterministic model

can still provide benefits even in the presence of certain types of uncertainties.

Thesis Supervisor: Hamsa Balakrishnan

Title: Associate Professor of Aeronautics and Astronautics

Acknowledgments

First of all, I sincerely thank my advisor, Professor Hamsa Balakrishnan, for her kind and consistent academic support for the longest time. She encouraged me to enter a doctoral program at MIT after getting my MS degree, guided my research in the right direction, and provided lots of advices on my work. She has also waited for me with patience when my research progress was stuck, I suffered from the aftermath of a four-car rear-end collision, I became a dad of two daughters, both my laptop and its backup hard drive were broken simultaneously, and I experienced difficulty in job seeking. Right after the car accident, she told me that life is more important than engineering. At that time, I was deeply impressed by her warmth and concerns on my health.

I would also like to thank my other two thesis committee members Prof. John Hansman and Dr. Richard Jordan for their interest, guidance, and feedback. I greatly appreciate Dr. Waqar Malik and Dr. Tom Reynolds for reviewing my thesis carefully and giving me helpful feedback. A special thank-you note should go to Dr. Yoon Jung at NASA Ames Research Center for his support and kindness. I am also grateful to Dr. James Kuchar, Dr. Allison Chang, Dr. Masha Ishutkina, and Lanie Sandberg of MIT Lincoln Laboratory for helpful discussions about my research.

This research work was supported by the National Science Foundation under grants 0931843 and ECCS-0745237. Much of the work related to the airport surface traffic scheduling problem in this thesis was motivated by the NASA Project NNX07AU34A, titled “Surface Traffic Optimization in the Presence of Uncertainties” under the direction of Principal Investigator Prof. John-Paul Clarke. So, I would like to express my gratitude to my colleagues in the project: Prof. John-Paul Clarke, Prof. Eric Feron, Dr. Leihong Li, Gustaf Solveling, Sanghyun Kim, and Bo-sung Kim of the Georgia Institute of Technology; and David Rappaport, Katy Griffin, Peter Yu, and Michelle Lu of Saab Sensis.

I appreciate every single members and alumni of ICAT, including Varun Ramanujam, Ioannis Simaiakis, Harshad Khadilkar, Lishuai Li, Yashovardhan Chati, Sathya

Silva, Alexandre Jacquillat, Tamas Kolos-Lakatos, and other good people for making the lab energetic. I am also thankful to my friends at MIT, including Korean Aero/Astro graduate students, Korean Film Festival staff, SSHS friends, and Tech Catholic Community people led by Fr. Richard Clancy, who have made my life in Boston enjoyable.

Lastly, I would like to thank my family. Many thanks to my wife, Rosa, who has supported me all the time and stayed with me whenever I felt happy, sad, pleasant, and frustrated. I could not have finished this long journey without you. I also thank my two pretty daughters, Angela and Helena, who wake me up at 6am every morning, make my life more valuable, and give me lots of delight. I am extremely grateful to my sister's family, sister-in-law, brother-in-law, and parents-in-law for their full material and emotional support. This PhD thesis is dedicated to my mother and grandmother who raised me with unconditional love. To my grandmother, I am really sorry for not being with you when you passed away. Rest in peace, my grandma.

Contents

1	Introduction	17
1.1	Motivation	18
1.2	Airport taxi processes	20
1.3	Literature review	22
1.3.1	Runway scheduling	22
1.3.2	Taxiway scheduling	24
1.3.3	Integration of planning tools in airport operations	28
1.3.4	Analysis of uncertainties in airport operations	29
1.3.5	Airside simulation models	31
1.4	Contributions of this thesis	33
1.4.1	Airport surface traffic optimization and simulation	33
1.4.2	Departure management strategies	35
1.4.3	Impact of uncertainty on surface operations	35
1.5	Organization of the thesis	36
2	Optimization architectures for taxiway and runway scheduling	39
2.1	Modeling assumptions	39
2.2	Integrated approach	40
2.2.1	Decision variables	40
2.2.2	Objectives	41
2.2.3	Constraints	41
2.2.4	Mathematical formulation	42
2.2.5	Strengths and weaknesses of the integrated approach	44

2.3	Three-step approach	45
2.3.1	Methodology	45
2.3.2	Step 1: Taxi-out time estimation for departures	46
2.3.3	Step 2: CPS algorithm for runway scheduling	47
2.3.4	Step 3: MILP model for taxiway scheduling	52
2.3.5	Expected benefits of the three-step approach	55
2.4	Evaluation of two optimization approaches with one-day traffic data .	55
2.4.1	Optimization set-up	56
2.4.2	Optimization result for one-day flight schedule	59
2.5	Departure planning method evaluation through fast-time simulation .	61
2.5.1	SIMMOD airfield model	62
2.5.2	Fast-time simulation result analysis	63
2.6	Conclusions	65
3	Case study at Detroit airport	67
3.1	Set-up for optimization and evaluation	67
3.1.1	Optimization cases for comparison	68
3.1.2	Airport performance metrics	69
3.2	Scenario 1: Constant high traffic demand with a fixed fleet mix ratio (5% heavy aircraft)	71
3.2.1	Traffic data for Scenario 1	71
3.2.2	Optimization results	71
3.2.3	Comparison with simulation results	75
3.2.4	Effect of the objective in runway scheduling	77
3.3	Scenario 2: Constant high traffic demand with a fixed fleet mix ratio (10% heavy aircraft)	79
3.3.1	Traffic data for Scenario 2	79
3.3.2	Optimization results	79
3.3.3	Comparison with simulation results	82
3.3.4	Effect of fleet mix ratio	84

3.4	Scenario 3: Cyclic high traffic demand with two peak times	85
3.4.1	More realistic traffic data for Scenario 3	85
3.4.2	Optimization results	86
3.4.3	Comparison with simulation results	92
3.4.4	Effect of traffic demand fluctuation	95
3.5	Conclusions	98
4	Comparison of individual aircraft trajectory-based control and aggregate queue-based control	101
4.1	Aggregate queue-based control for departure management (N-Control)	102
4.2	Individual aircraft trajectory-based control	105
4.2.1	RTA control	105
4.2.2	PbT control	106
4.3	Implementation of different departure control approaches	107
4.3.1	Implementation of N-Control	107
4.3.2	Implementation of RTA control and PbT control	109
4.4	Airport performance comparison	110
4.4.1	Gates	111
4.4.2	Taxiways	113
4.4.3	Runways	117
4.5	Gate conflicts	119
4.5.1	Gate conflict frequency	120
4.5.2	Options to resolve gate conflicts	121
4.6	Conclusions	125
5	Impact of uncertainty on airport performance and planning	127
5.1	Uncertainty in airport operations	127
5.2	Simulation modeling for uncertainty studies	128
5.2.1	Simulation model validation	128
5.2.2	Flight schedules used in simulations	130
5.2.3	Iterations with random variables	131

5.3	Pushback time variations	132
5.3.1	Pushback time uncertainty study at current traffic level	132
5.3.2	Pushback time uncertainty study at higher traffic level	134
5.3.3	Impacts of pushback time uncertainty on gate conflicts	137
5.4	Landing time variations	137
5.5	Taxi speed variations	138
5.5.1	Case study 1: Taxi speed variations on taxiways only	139
5.5.2	Case study 2: Taxi speed variations on taxiway and ramp areas	141
5.5.3	Case study 3: Taxi speed variations on taxiway and ramp areas, with faster arrivals	143
5.6	Inter-departure time variations	145
5.7	Conclusions	147
6	Summary and Next Steps	149
6.1	Summary of this thesis	149
6.2	Future research directions	151
6.2.1	Extensions of airport surface traffic optimization	151
6.2.2	Adaptations to other airports	153
6.2.3	Applications of fast-time air traffic simulations	154

List of Figures

1-1	Airport taxi processes	21
2-1	Sequential process diagram for three-step approach	46
2-2	CPS network example for $n = 5, k = 1$	50
2-3	DTW airport layout (as of 2007)	56
2-4	Node-link network model for DTW	57
2-5	SIMMOD airfield model for DTW	62
2-6	Total taxi-out time and gate-holding time for departures	65
2-7	Total taxi-in time for arrivals	65
3-1	Scenario 1: Average taxi-out time per aircraft	72
3-2	Scenario 1: Average taxi-in time per aircraft	72
3-3	Scenario 1: Average takeoff delay per departure	73
3-4	Scenario 1: Takeoff order changes from initially estimated sequence	74
3-5	Scenario 1: Takeoff order changes between Step 2 and Step 3	74
3-6	Scenario 1: Total runtimes	75
3-7	Scenario 1: Average taxi-out time comparison between optimization and simulation	76
3-8	Scenario 1: Takeoff order changes between optimization and simulation	76
3-9	Scenario 1: Average taxi-in time comparison between optimization and simulation	77
3-10	Scenario 2: Average taxi-out time per aircraft	80
3-11	Scenario 2: Average taxi-in time per aircraft	81
3-12	Scenario 2: Average takeoff delay per departure	81

3-13 Scenario 2: Takeoff order changes from initially estimated sequence . . .	81
3-14 Scenario 2: Takeoff order changes between Step 2 and Step 3	82
3-15 Scenario 2: Total runtimes	82
3-16 Scenario 2: Average taxi-out time comparison between optimization and simulation	83
3-17 Scenario 2: Takeoff order changes between optimization and simulation	83
3-18 Scenario 2: Average taxi-in time comparison between optimization and simulation	84
3-19 Daily traffic demand pattern at DTW in 2011	86
3-20 Scenario 3: Average taxi-out time per aircraft	87
3-21 Scenario 3: Average taxi-in time per aircraft	87
3-22 Scenario 3: Average takeoff delay per departure	88
3-23 Scenario 3: Takeoff order changes from initially estimated sequence . .	88
3-24 Maximum takeoff delay - optimization	89
3-25 Scenario 3: Takeoff order changes between Step 2 and Step 3	89
3-26 Scenario 3: Runway throughput for both departure runways	90
3-27 Scenario 3: Number of taxiing aircraft for each departure runway . .	91
3-28 Scenario 3: Runway throughput vs. number of taxiing aircraft	91
3-29 Scenario 3: Total runtimes	93
3-30 Scenario 3: Average taxi-out time comparison between optimization and simulation	93
3-31 Scenario 3: Takeoff order changes between optimization and simulation	94
3-32 Scenario 3: Average taxi-in time comparison between optimization and simulation	94
3-33 Maximum takeoff delay - SIMMOD result	95
3-34 Scenario 3: Aircraft flow comparison	96
4-1 Typical airport performance curve for takeoff rate as a function of the number of aircraft taxiing out	103
4-2 Average gate-holding time and number of gate-held flights	111

4-3	Taxi-out time comparison	114
4-4	Taxi-out time comparison for each departure runway	115
4-5	Taxi-in time comparison	115
4-6	Number of taxiing aircraft for each departure runway	116
4-7	Number of stops during taxiing	117
4-8	Runway throughput for both departure runways	118
4-9	Runway throughput vs. number of taxiing aircraft	118
4-10	Maximum takeoff delay	119
4-11	Gate conflict frequency	121
4-12	Total gate-in delays by departure runway	123
4-13	Gate-in delays by departure runway	123
5-1	Taxi-out time comparison between SIMMOD simulations and surveil- lance data	130
5-2	Unimpeded taxi-out time comparison between SIMMOD simulation and queueing model	130
5-3	Average ground delay per departure with pushback time perturbation at current traffic level	134
5-4	Average ground delay per departure with pushback time perturbation at higher traffic level	135
5-5	Average ground delay per arrival with pushback time perturbation at higher traffic level	136
5-6	Gate conflict frequency by pushback time uncertainty at higher traffic level	138
5-7	Average ground delay with taxi speed perturbation on taxiway area at current traffic level	140
5-8	Average ground delay with taxi speed perturbation on taxiway area at higher traffic level	141
5-9	Average ground delay with taxi speed perturbation on taxiway and ramp area at current traffic level	142

5-10	Average ground delay with taxi speed perturbation on taxiway and ramp area at higher traffic level	143
5-11	Average ground delay with taxi speed perturbation on taxiway and ramp area with faster arrivals at current traffic level	144
5-12	Average ground delay with taxi speed perturbation on taxiway and ramp area with faster arrivals at higher traffic level	144
5-13	Average ground delay with separation time perturbation at current traffic level	146
5-14	Average ground delay with separation time perturbation at higher traffic level	146

List of Tables

2.1	Possible aircraft assignments for $n=5$, $k=1$	49
2.2	Minimum separation (in seconds) between takeoffs	58
2.3	Optimization result for 1-day flight schedule	59
2.4	Total runtime for 1-day flight schedule	61
3.1	Optimization cases for comparison	68
3.2	Airport performance metrics	69
4.1	N-Control approach gate-holding result depending on N_{ctrl}	112
4.2	N-Control approach gate-holding result for Runway 22L	112
4.3	N-Control approach gate-holding result for Runway 21R	113
4.4	Effects of early pushback of departures	124
5.1	Impact of pushback time uncertainty on the ground delay	133

Chapter 1

Introduction

Flight delays are recognized as one of the main obstacles to the steady growth of the air traffic demand. According to the report issued by the Joint Economic Committee (JEC) of the U.S. Senate, the total cost of domestic air traffic delays to the U.S. economy was estimated to be about \$41 billion in 2007 alone. The JEC report also stated that 58%, 20%, and 8% of flight delays occurred at the gates, taxiing out to the runway, and taxiing into the gates upon landing, respectively, while only 15% of the total delays were airborne delays [72]. Delays on the airport surface increase airline operating costs and environmental impacts because the amount of fuel burned on the surface is approximately proportional to the taxi times of aircraft [77,115]. Airport congestion can also result in increased controller workload and safety concerns [76].

Airport congestion may be mitigated through one of the following approaches: (1) reorganizing flight schedules to reduce traffic demand during peak periods (demand management); (2) increasing the airport capacity by expanding airport resources such as gates, runways, and taxiways; and (3) using available airport capacity more efficiently by improving air traffic operations both on the surface and in the surrounding terminal areas [123].

Rescheduling flights and expanding infrastructure are both difficult in practice, and can be expensive. It is also believed that inefficient operations lead to airport congestion and associated delays [17,109]. Therefore, this thesis focuses on techniques to improve airport surface operations.

1.1 Motivation

Airport surface operations include taxiway and runway operations. Traditionally, optimization models for taxiway scheduling (determining the optimal pushback times of departures, gate-in times of arrivals, and passage times of individual taxiing aircraft at control points on the ground to minimize taxi times) and for runway sequencing/scheduling (determining the optimal takeoff sequence and times over each departure runway to improve the efficiency of runway use) have been independently developed, and then linked as separate modules [16,59,92]. However, since the two elements of surface operations are closely related at an airport, greater benefits may be achievable if they are considered simultaneously [8].

Several optimization models have been developed to calculate optimal taxi schedules, subject to operational and safety constraints. These optimization models have shown the promise of taxi times savings when applied to busy airports. However, they are currently not amenable to real-time implementation during peak times because of their slow computational performance [85,104,107]. They also assume that either a scheduled takeoff time or a target time for departure is given by another tool such as a *Runway Scheduler* [16] or a *Taxi-out Time Estimator* [59].

The runway has been identified as the main bottleneck in airport operations [69]. Runway scheduling algorithms have been proposed for maximizing runway throughput, considering wake vortex separations depending on aircraft weight classes [4,13,19]. These approaches focus on the runway sequencing and separation between aircraft only, but do not consider the interaction with taxiway conditions and the impact of arrivals on the taxiway and ramp areas, when scheduling the optimal takeoff times of departures. The optimal runway schedule without accounting for taxiway operations could potentially have an adverse effect on taxiway operations, such as long waiting times in the departure queue [59].

The main objective of this thesis is to develop a fast and efficient algorithm for both runway and taxiway scheduling by bridging the gap between the two scheduling problems. The system-optimal solution from a unified model of airport operations

may have more benefits, compared to the solution obtained by sequentially connecting the separate optimization models. Similarly, when optimizing both runway and taxiway schedules, the combined optimal solution from an integrated approach may be better than the solution from a step-by-step approach, in which the runway schedule is optimized first, and then the taxiway schedule is optimized with the target takeoff times determined in runway scheduling, because the runway schedule can be affected by taxiway conditions. To use a proposed scheduling algorithm in practice as a decision support tool, its computational performance and tractability are also important characteristics.

From a practical point of view, following the optimal runway and taxiway schedules precisely is not realistic at the present level of technology because the time-based control of individual aircraft trajectories using the schedule optimization algorithm requires advanced equipment and procedure changes. This microscopic control approach is suitable for a long-term strategy to accommodate the increased air traffic demand in the future. To mitigate the current surface congestion, we need a near-term control approach. Some departure control methods like N-Control have been developed and tested at several busy airports with little procedure modification [95,116]. These macroscopic methods manage the traffic congestion level on the surface below a threshold by holding some departures at gates.

While planning the efficient airport surface traffic movements, it is also necessary to account for uncertainties. Due to the presence of considerable uncertainty in flight readiness, taxi processes, and runway operations, the actual movement of taxiing aircraft can be significantly different from the predictions of optimization models [26]. Consequently, the actual ground delay is usually greater than the estimated delay from the models, because the optimization approaches typically deal with deterministic settings. This tendency may also result in unnecessarily excessive gate holding when the optimization models are applied [87].

Researchers studying airport system management and planning have recognized the presence of uncertainty in airport operations and its importance, but detailed investigations of the uncertainties at the microscopic level have not been conducted

to date. It is only in recent years that the variability in the movement of taxiing aircraft has been analyzed using flight data [10,28,29]. Moreover, it is difficult to develop a stochastic optimization model because a large number of scenarios can exist even for a small number of flights [26].

Over the past few decades, many simulation tools have been developed for modeling airport operations and analyzing the related statistics [39]. Among them, microscopic simulation models like SIMMOD have been enhanced to simulate the movement of individual aircraft [98]. These models can also simulate stochastic processes using random variables, to reflect the uncertainty in airport operations. We therefore propose to use a simulation tool (SIMMOD) for generating various scenarios related to the uncertainty, such as pushback time and taxi speed, and investigating the impacts of these sources of uncertainty on delays.

1.2 Airport taxi processes

When a departing flight is ready to pushback and receives pushback clearance from ramp control, the aircraft is pushed out of the gate area. After its engines are started and the towbar is disconnected, a guide crew member confirms that the ramp area is clear to taxi. The pilot then contacts ground control to get taxi clearance and routing to the active runway. In case that there exist spots between gates and taxiways at the airport, the pilot first contacts ramp control to get taxi clearance up to a spot near the gate. On reaching the spot, the pilot contacts a ground controller and gets clearance for taxi. Once taxi clearance is received, the pilot starts taxiing the aircraft out with visual checks. In some cases (for example, long taxi routes), single engine taxi procedures may be used for reducing fuel burn and emissions. Depending on the airport layout, taxiing aircraft may have to wait before crossing active runways. On reaching close to the departure runway, the pilot switches frequency to local controller channel and follows instructions from local control regarding takeoff. An aircraft may experience delays before its takeoff if there are many departures ahead of it in the departure queue.

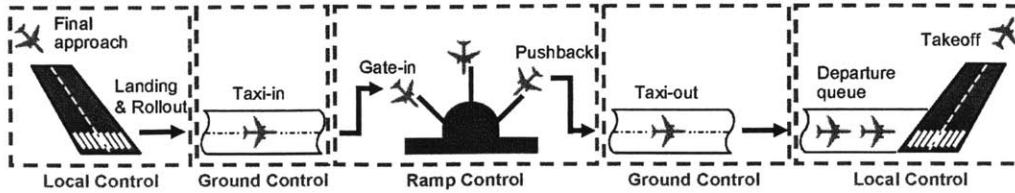


Figure 1-1: Airport taxi processes

Arriving aircraft follow a similar process, with the order reversed. After touching down on the runway, the aircraft decelerates to taxi speed and vacates the runway. To reduce the runway occupancy time and maximize runway capacity and safety, many airports operate high-speed runway exits. After exiting the runway, the pilot contacts ground control for taxi-in instructions, completes the “after landing taxi” checklist, and calls the local ramp control to confirm the arrival gate assignment and occupancy status. If the arrival gate is occupied by another aircraft, the arrival may have to wait at a remote location until the gate is clear. After receiving clearance to the gate, the aircraft moves into the ramp area and stops at a designated point for gate access [21]. These taxi processes are illustrated in Figure 1-1. It is important to note that while taxiing, departures and arrivals move on the surface simultaneously, share airport resources such as terminals and runways, and interact with each other at the airport.

Many researchers have been developing new operating concepts and procedures to enhance the efficiency of the airport system in preparation for increasing air traffic demand. It is expected that in the near future, taxi operations will experience a shift from voice to data link. Instead of voice communications between controllers and pilots, taxiing aircraft on the ground will be guided by time-based or speed-based taxi clearances via data link at various traffic flow points throughout the taxi routes to regulate the required precision of aircraft movements on the ground. This new surface traffic management system and its operating concepts can be realized with the support of advanced technologies such as surface maps, head-up and head-worn displays, four-dimensional trajectory (4DT)-based guidance algorithms, and traffic conflict detection and alerting systems [101]. These technologies will improve the

accuracy of prediction of aircraft movements on the ground, and make it possible to control the surface traffic more efficiently without loss of safety.

1.3 Literature review

Previous research about the aircraft movement scheduling in airport operations can typically be divided into two categories: runway scheduling and taxiway scheduling. Most of the optimization models reviewed in Section 1.3.1, 1.3.2 and 1.3.3 are based on deterministic settings. Literature dealing with the uncertainty in airport operations is reviewed in Section 1.3.4. These uncertainties can be modeled by simulation tools at a microscopic level. Therefore, we also review the general characteristics of air traffic simulation tools in Section 1.3.5, in order to investigate how airport and airspace simulation tools such as SIMMOD can be used in our research.

1.3.1 Runway scheduling

Various approaches have been proposed for solving the aircraft sequencing and scheduling problem for runways and terminal areas. The goal of runway scheduling is to find the optimal takeoff or landing schedule that simultaneously achieves safety, efficiency, and equity, which are often competing objectives [5,25,32]. In addition to optimizing multiple objectives, modeling the runway scheduling problem to find the optimal solution in a reasonable amount of time has remained a challenge. One reason for this computational hurdle is that most runway scheduling models are, from a theoretical perspective, inherently hard to solve [19]. As a result, most practical approaches rely on heuristic or approximate algorithms to obtain “good” solutions within reasonable computation times [4,25,38].

Considering limited flexibility in reordering and fairness to airlines, Dear proposed the Constrained Position Shifting (CPS) method, in which deviations from the First-Come, First-Served (FCFS) order are limited [40]. Based on the CPS approach, a heuristic algorithm for single runway scheduling was presented. This algorithm considered separation requirements and the maximum position shift (MPS) parameter,

enforcing the constraint that an aircraft cannot be shifted by more than this parameter from the FCFS order [41]. Later, a variety of scheduling algorithms using the CPS concept were evaluated and statistically analyzed under different scenarios [46,96].

Some researchers have modeled the aircraft sequencing problem as a job shop scheduling problem by regarding runways and aircraft as machines and jobs, respectively [19,23,24,102]. However, due to separation requirements between aircraft, the processing time of a job on a machine depends on the previous job on the same machine. Therefore, the aircraft sequencing problem is a special case of the job shop scheduling problem, with sequence-dependent processing times and time windows. Psaraftis [102] incorporated the CPS concept within a dynamic programming recursion for solving the aircraft arrival sequencing problem at a single runway as a special case of the job shop scheduling problem. Although the problem could be solved in polynomial time, time window restrictions for landing and precedence relationships among flights were not taken into account. Venkatakrisnan et al. [124] modified Psaraftis' formulation in a heuristic manner to consider the earliest and latest times when they investigated the separation times observed between landings at Boston Logan airport (BOS). Trivizas [122] proposed a dynamic programming approach to compute the optimal CPS landing sequence, but time window restrictions and precedence relations between aircraft were not considered.

There have also been several attempts to apply integer programming techniques to the problem. Bianco et al. [23,24] adopted a job shop scheduling view for the aircraft sequencing problem and solved the single runway landing problem using a Mixed-Integer Linear Programming (MILP) model. Abela et al. [1] presented a binary mixed integer formulation of the single runway aircraft landing problem, together with a heuristic based on a genetic algorithm. Beasley et al. [19,20] extended this MILP model to the case of both single and multiple runways. With the integer programming method, they could reflect constraints such as time windows, precedence relations, and limits on the maximum number of position shifts, but the solution times were often too long to utilize the method as a real-time decision support tool. Ernst et al. [45] developed a fast simplex-based lower-bounding method for the aircraft scheduling

problem and used it for solving single and multiple runway problems with a heuristic, as well as with an exact branch-and-bound method.

Bayen et al. [18] formulated the aircraft sequencing problem as a single machine scheduling problem and presented approximation algorithms to alternatively minimize the sum of delays and the landing time of the last aircraft in the sequence. The approximation algorithm was slower than a heuristic algorithm, but provided guarantees on sub-optimality and performed more robustly for a range of sequences than a greedy heuristic algorithm [108].

Recently, Balakrishnan et al. [13,14,15,33] posed the runway scheduling problem as a modified shortest path problem on a network and solved it with a dynamic programming algorithm under the CPS framework. They showed that their approach could handle operational constraints that may arise in practice, and that its computation time was sufficiently short to enable real-time implementation. This approach was extended to the problem of runway scheduling with a variety of objectives [82,83,84].

While most of the algorithms mentioned above were basically developed for arrivals, there has been less research focused on the departure runway scheduling problem. The dynamic programming-based approach to sequencing landing slots developed by Balakrishnan et al. could be applied to departure scheduling with little modification [14,15,82]. MILP models to optimize the sequence and schedule of departures were also proposed for Dallas/Fort Worth airport (DFW), considering its local features such as multiple departure queues and runway crossings [60,61,89]. Anagnostakis et al. suggested a two-stage approach to solving the departure sequencing problem where the first stage determined a departure sequence based on aircraft weight classes only and the second stage assigned individual flights to the sequence [3]. This approach was extended to the stochastic departure runway planning problem to obtain more robust sequences in the presence of uncertainties [119].

1.3.2 Taxiway scheduling

There have been several efforts on improving the efficiency of airport surface operations, mitigating congestion level on the taxiway and ramp area, and reducing taxi

times, fuel burn, gas emissions and noise level on the ground. As part of the development of the Departure Planner, there were comprehensive discussions on air traffic flow restrictions in the terminal area and potential control points for surface operations [49]. Based on these discussions, runways were considered as the limiting factor for airport capacity [69], and taxi-out times were estimated using a queuing model [68,103].

Simaiakis and Balakrishnan developed a predictive queuing model to estimate the taxi-out times from gates to the departure runways by including the effect of taxiway interactions [114]. They also used this model to evaluate the potential reduction in taxi times, fuel burn and emissions from queue management strategies [115]. In this control approach, the traffic flow on the surface was managed in an aggregate manner. Field tests at BOS airport demonstrated that the departure control strategy based on this queuing model could achieve significant benefits in the current operational environment with minimal procedural modifications and additional controller workload [113,116].

As an alternative approach to managing the surface traffic on the basis of individual aircraft, on the other hand, some researchers focused on the development of the optimization algorithms to solve the Aircraft Taxi-scheduling Problem (ATP) [16, 85,104,117,125]. The objective of the ATP problem is to minimize the taxi times of individual flights and mitigate traffic congestion on the surface, subject to operational rules and safety concerns. The optimization model for this problem determines the optimal times for each aircraft to leave its gate or runway exit and reach significant control points along its taxi route, while considering the movements of the other flights on the ground. For minimizing taxi times, either a gate-holding strategy or alternative taxi routes are generally used. Solving this problem corresponds to a microscopic approach to surface traffic management, which controls individual aircraft trajectories on the surface. Such an approach uses a node-link network model to represent the airport layouts.

Some prior approaches for the airport surface traffic optimization include Dynamic Programming-based taxi route optimization using Dijkstra's algorithm [34] and Time-

Dependent Shortest Path techniques [121]. Visser and Roling [107,125] also proposed a MILP model for a taxi movement planning tool. However, most examples demonstrated in these efforts were limited to ideally modeled, small size network models, and were not amenable to be used as a real-time decision support tool.

Several researchers have studied surface traffic optimization problems using actual airport models and flight schedule data. Smeltink et al. [117] developed a MILP model to determine the movement of taxiing aircraft and meet basic safety and operational constraints for simulations of the Amsterdam airport node-link model, using rolling horizon algorithms to accommodate uncertainties. The model, however, had long solution times and did not consider some factors such as runway occupancy times and safety constraints. Roling also tested his taxi planning system based on the MILP model with realistic peak day flight schedules at Amsterdam Airport Schiphol (AMS) [105] and Hartsfield-Jackson Atlanta International Airport (ATL) [106]. Rathinam et al. [104] improved Smeltink et al.'s MILP model and applied their approach to simulations at DFW airport. They tried to consider as many operational constraints as possible, including the aircraft types for separations on the taxiway, but the model, tested for the real-world scenarios with departures only, showed long computation times for high-density traffic. Their model was recently extended for solving a Multiple Route Aircraft Taxi Scheduling Problem (MRATSP) by introducing routing decision variables [94].

Balakrishnan and Jung [16] proposed an Integer Programming (IP) formulation for optimizing surface operations at DFW airport by adapting the Bertsimas-Stock Patterson formulation for the Air Traffic Flow Management (ATFM) problem [22]. Through simulations with actual DFW airport data, they evaluated two strategies for improving the taxi times: controlled pushback (gate-holding strategy) and taxiway reroutes. This model improved the formulation for the surface traffic optimization and its computational performance, but did not account for several operational restrictions such as overtaking constraints and collision avoidance at intersections. This model was further improved by taking these restrictions into account [85]. By dividing long taxiway links into several pieces and limiting their capacity up to one aircraft,

overtaking on the taxiway could be prevented. Moreover, safety constraints such as head-to-head collision avoidance and head-to-tail collision avoidance on the ground were added to the model. In order to make the model more practical, existing flights that were already moving on the ground at the beginning moment of the optimization were also included in the model as parameters. However, this model showed poor computational performance at high density traffic because a number of variables were generated by link segmentation for the overtaking restriction. Frankovich [51] applied a similar IP model to historic data at DFW and BOS airports and showed significant benefits with good computational tractability, although the optimization model was based on simplified network graphs for the airport layouts and did not consider overtaking constraints on the surface.

As an alternative approach, heuristic methods have been applied, exclusively by using Genetic Algorithms (GAs). GAs do not guarantee the optimality of the solutions, but show shorter solution times, which can sometimes compensate for the sub-optimality. In the GA approaches, timings and routings of the aircraft ground traffic are optimized to avoid conflicts using crossover and mutation operators [56,57,58,100]. A two-phase approach based on the genetic algorithms has also been investigated, which considers the runway sequencing first and the ground movement in the second stage [42,43].

Algorithms for taxiway scheduling should reflect the dynamic nature of the airport system [8]. When the algorithms to solve the ATP problem are implemented in the real situations at airports, they typically follow a rolling horizon procedure. In other words, the prediction of aircraft taxi schedules is periodically updated by re-optimization with new information about the next planning horizon. The use of a rolling horizon method can not only accommodate the inherent dynamic nature of the system, but can also reduce the computational complexity of the ATP problem [117]. In addition, we may obtain more robust taxi schedules since some uncertainties in the taxi process are removed as time goes on. However, we cannot ensure that the solution from a series of local sub-problems is the global solution obtained by solving the whole problem at once.

In the rolling horizon problem, it is important that the time window for optimization (the planning horizon) is set within a reasonable range, accounting for both global optimality and computational performance. If the time window used for optimization is too short, we may obtain myopic short-term solutions that do not consider the taxiing aircraft in the next time window. The use of a too large time window may need a significant computation time to find an optimal solution [36]. Practical implementations use 15 minutes as a typical horizon, taking the average taxi time and computational feasibility into account [86,113,117].

1.3.3 Integration of planning tools in airport operations

Runway and taxiway scheduling in airport operations cannot work in isolation because they are closely linked with each other [9]. In general, however, the optimization models embedded in these planning tools have been developed independently. If the taxiway schedule is optimized through the integration of the sequence and schedule of departures and arrivals over runways, more benefits would be expected. However, research about this integration has not been done much so far due to its complexity [8].

Departure sequencing is sometimes included in the optimization models for taxiway scheduling [36,42,43,76]. In this integrated modeling approach, however, the target times of departures are given, and the models just ensure that the departures satisfy wake vortex separation requirements. Furthermore, the proposed models mainly focus on minimizing the overall taxi times, rather than optimizing the takeoff times as well.

Instead of optimizing different types of operations simultaneously, coordinating the separate planning modules has also been suggested [16,37,51,74,89,90,92]. In this approach, the runway sequence and schedule is optimized first, and taxiway scheduling is then performed using the optimal takeoff times. Such sequential planning makes it possible to connect the independent optimization components in the integrated system with common data.

Spot And Runway Departure Advisor (SARDA), which NASA has developed to improve the efficiency of surface operations through Air Traffic Control Tower (ATCT)

advisories, also follows the sequential approach [74]. The SARDA scheduler is based on the Spot Release Planner (SRP) [89,90], a method to provide metering advisories. SRP is a two-stage algorithm. The first stage is a Runway Scheduler (RS) [61,93] which provides the optimal runway schedule including takeoff times for departures and crossing times for arrivals. The second stage determines optimal release times from assigned spots or gates to meet the optimal departure schedules. The tactical gate hold method using SARDA were tested for the east side of DFW with various traffic scenarios both in an automated simulation environment and human-in-the-loop (HITL) experiments, and the results showed significant reduction in taxi delay and fuel consumption without increasing controller workload [63,64,65]. Fast-time simulations at different airports such as Philadelphia International Airport (PHL), Charlotte-Douglas International Airport (CLT) and Los Angeles International Airport (LAX) also demonstrated that the SARDA concept could provide substantial benefits at these airports as well [11].

1.3.4 Analysis of uncertainties in airport operations

In general, there is considerable uncertainty in airport operations. The uncertainty arises from differences in flight readiness, pushback processes, taxi speeds, pilot-controller communications, etc. Irregular events such as mechanical problems and safety incidents also contribute to uncertainty. These factors can produce variability in the earliest possible time of pushback, departure sequence, takeoff/landing times, passage time at each intersection on a taxi route, crossing time at a departure/arrival fix, and departure/arrival spacing [26].

Analysis of surveillance data using the Surface Operations Data Analysis and Adaptation (SODAA) tool has shown the impacts of uncertainty on airport surface operations [10,28,29]. This research demonstrated the variability observed in current surface operations, specifically runway occupancy times, taxi times around corners, time to reach runway crossings, runway crossing times, and taxi paths actually used. The variability in the actual movements of taxiing aircraft may also make it hard to follow planned four-dimensional trajectories (4DTs) in the future.

Statistical approaches using queueing theory or regression techniques can be used to better reflect uncertainty [73,114]. Particularly, in the prediction of taxi times, the variability in the unimpeded taxi-out time can be taken into account using the expected value and standard deviation [115]. When the pushback time is actively controlled, the queue-based aggregate model shows a higher taxi-out time, quite close to the actual value, compared to the individual flight trajectory-based optimization model for taxiway scheduling. This difference arises because the optimization model is not able to appropriately account for uncertainty [87].

Stochastic optimization models of airport operations are relatively few in number. A probabilistic approach was introduced in [26], but required the enumeration of a large number of scenarios for representing the variability. Gotteland et al. [58] modeled the aircraft taxi speed uncertainty as a fixed percentage of the predefined speed in their GA approach, but other factors such as pushback times were still assumed as deterministic. Most recently, Anderson et al. [6] included uncertainty in their formulation for taxiway scheduling, but their MILP model was tested only for a simple taxiway topology because of its complexity. Their optimization program considered the uncertainty in aircraft taxi speeds, gate pushback times, and stopping times in the constraints in the form of Gaussian distributions, and determined the optimal flight schedules to minimize the probability of constraint violation, as well as the total taxi time.

Several researchers have focused on the departure runway scheduling in the presence of uncertainties because the perturbations accumulated from various uncertainty factors in the airport taxi process manifest themselves at runways, which are the main bottleneck to determine the airport capacity. Solveling et al. [118,119] developed a stochastic runway planning model addressing the uncertainties from departure pushback delay, taxiing delay, and arrival prediction error by extending the two-stage deterministic algorithm for runway operations planning proposed by Anagnostakis et al [3]. In addition, the Integer Programming (IP) model for the airport operations optimization problem proposed by Frankovich was also extended to incorporate the key uncertainties in runway availability and in the earliest possible runway times of

flights [51].

Instead of developing the stochastic optimization model, Gupta et al. [62] evaluated the impacts of uncertainty on a deterministic model for runway scheduling. They tested their MILP model with various traffic conditions and different levels of uncertainty in earliest readiness for takeoff or arrival crossing, and showed that the deterministic approach could achieve better performance than a FCFS policy, even in the presence of uncertainty.

1.3.5 Airside simulation models

Many simulation tools have been developed for the analysis of airport airside operations. These models can be categorized as *macroscopic* or *microscopic*, depending on the level of modeling detail. Macroscopic models are mainly analytical in nature (e.g., FAA Airfield Capacity Model, DELAYS) and cover the operations at runways and final approaches. They can be used to compute the airport capacity and the cost of flight delays for policy analysis, cost-benefit studies, and approximate traffic flow analysis. On the other hand, the microscopic simulation models reflect individual aircraft movements and conflicts with other aircraft, and deal with more tactical issues in runway and taxiway operations, as well as in terminal area airspace. These models are built based on a discrete-event simulation approach, where system states change only at the moments when certain events occur. Examples of such tools are SIMMOD, TAAM, The Airport Machine, RAMS, and HERMES. This type of simulation software can be used for the detailed traffic flow analysis, as well as for the preliminary design of new airport layout and procedure. Most microscopic models represent the airfield and airspace as a network of nodes and links, and aircraft in the simulations follow prescribed paths on this network [39,97,98]. Time-based simulation models like VTASIM have also been developed to represent dynamic movements of aircraft, such as changes of aircraft speed [12,121].

SIMMOD is the most well-known airspace and airfield simulation model, capable of calculating airport capacity, flight travel time, delay and fuel consumption [2,7]. This tool can build airspaces and airports from input data, simulate detailed traffic

flows, and generate reports of all outputs needed for the study. The input data consisting of aircraft, airspace, airfield and event information address ATC policies and procedures, physical layouts for airport and airspace, and flight schedules. For the analysis of simulation results, SIMMOD provides detailed outputs for each flight, and the related statistics. Output data include aircraft travel times, traffic flows at specific points, capacity, delays and their reasons, and fuel consumption.

SIMMOD has been enhanced under the FAA's funding and validated using a number of case studies for real airports and airspaces [44,52,53,91,120]. This tool can be used to plan potential improvements by playing out alternatives in operations, technologies, or facilities through fast-time simulations. It can help make decisions at the tactical and strategic levels, and improve decision-making. Case studies include changes in airport layouts, runway operations or airfield ground operations, terminal traffic estimation, runway occupancy time estimation, and multi-airport systems like the New York area [44,91,120].

SIMMOD also supports stochastic simulations through repeated runs with random seeds. In order to generate realistic and statistically significant results from given inputs, it is necessary to run numerous iterations with randomized variables for a single data set. The random variables available in airport and airspace simulations include gate occupancy times, injection times of multiple arrivals and departures, takeoff and landing roll distances, airspace separations, delays, pushback or power-back times, runway crossing times, and slot window times. This function has been previously used in several case studies, including the evaluation of optimization algorithms to minimize air traffic delay costs [53,54,67,80,81].

SIMMOD can accept input parameters that are probabilistic quantities and capture the impacts of uncertainty on the chosen performance metrics. This capability enables us to study the effects of uncertain factors in airport operations on ground delay.

1.4 Contributions of this thesis

This thesis focuses on developing optimization models for airport surface traffic management, and analyzing the effects of different flight control approaches and sources of uncertainty on airport performance. The main contributions of this thesis include:

1. The development of two potential architectures for optimizing runway and taxiway schedules: a unified optimization model and a sequential approach connecting independent optimization modules.
2. The analysis of the impact of surface traffic optimization on airport performance metrics using various traffic scenarios at a busy airport with multiple runways.
3. The development of a fast-time air traffic simulation model using SIMMOD for observing how the proposed optimization models work in current operational environment.
4. The comparison of two departure control approaches for surface management: aggregate queue-based control and individual aircraft trajectory-based control.
5. The analysis of the effects of uncertainty sources in airport operations on airport performance through stochastic simulations using SIMMOD.
6. The investigation of the robustness of deterministic surface traffic optimization in the presence of uncertainty through stochastic simulations using SIMMOD.

These contributions are briefly described in the following sections.

1.4.1 Airport surface traffic optimization and simulation

This thesis proposes two airport operations optimization models for taxiway and runway scheduling. First, we propose a unified MILP model to optimize both runway schedule and taxiway schedule simultaneously. This optimization model is designed to minimize both runway delays and total taxi times by controlling pushback times at assigned gates and passage times at intersections on taxiways, while keeping various

operational constraints and safety concerns in surface traffic operations. The formulation is built based on a node-link network model that represents the airport layout including gates, taxiways, and runways. We also introduce a sequential approach that coordinates a runway scheduling algorithm with a taxiway scheduling model harmoniously, similar to approaches suggested by prior researchers [16,37,51,74,89,90,92]. In this optimization method, we first estimate the taxi-out times of departures, then determine the optimal takeoff sequence using a runway scheduling algorithm, and finally optimize the detailed taxiway schedule of each aircraft using a separate MILP model.

We compare these two approaches to optimizing airport surface traffic operations using actual flight schedule data at Detroit airport (DTW), which has multiple runways and requires careful control on the ramp areas around terminal buildings. We evaluate the performance of the optimal solutions and their computational properties in order to gauge which optimization approach is more suitable for real-time decision support tools at current traffic levels.

These surface traffic optimization methods are also applied to high density traffic scenarios expected to arise in the future. We compare the two optimization approaches with respect to various metrics representing airport performance, and assess the benefits of the gate-holding strategy for departure planning. For the sequential approach, we test various runway scheduling algorithms having different objectives (e.g., maximizing runway throughput and minimizing runway delays) to analyze their benefits compared to the current discipline on a FCFS basis and investigate their impacts on the eventual airport performance after taxiway scheduling. The effects of air traffic demand characteristics such as aircraft fleet mix ratio and demand fluctuation are also analyzed through comparisons of the optimization results from different traffic scenarios.

We also implement fast-time simulations using SIMMOD to evaluate the proposed optimization models. From a control point of view, the SIMMOD simulations are similar to current airport operations based on voice communications between air traffic controllers and pilots, because flights can be controlled only at several entry

points into the system like gates and arrival fixes, and not at every intersection point on the route. Through the SIMMOD simulations, therefore, we can observe how the optimization models work in current operational conditions. We first validate the simulation model with the historic flight schedule data, and then extend the fast-time simulations to evaluate various optimization cases for high demand scenarios.

1.4.2 Departure management strategies

There are two possible approaches to surface traffic management: individual aircraft trajectory-based control and aggregate queue-based control. This thesis compares these two different departure control approaches. These control approaches are implemented with the same traffic scenarios for comparison and evaluated in terms of various airport performance metrics. The simulation results tell us that at the high traffic level, the trajectory-based control approach can provide significant taxi time savings even in the current operational environment controlling pushback times at gates only, although the workload of ramp controllers may increase due to aggressive departure control. On the other hand, it seems that the amount of taxi time savings from the aggregate queue-based control is dependent on the capacity limit in departure queues. For the same flight schedule data, therefore, we conduct the fast-time simulations with a range of queue capacity control parameters and analyze the effects of the control parameter.

The gate-holding strategy commonly used in these departure control methods can cause gate conflicts between a gate-held departure and an arriving aircraft assigned to the same gate. We analyze the frequency of gate conflicts in a given traffic scenario and investigate its impacts on surface traffic. We also suggest several possible strategies to minimize the gate conflicts when a departure control approach is applied.

1.4.3 Impact of uncertainty on surface operations

Finally, this thesis uses stochastic simulations to evaluate the impacts of uncertainty on airport performance. We first identify various sources of uncertainty in airport

operations that influence the airport system performance. These uncertainty factors, such as pushback times, runway exit times, taxi speeds, and runway separation times, are embodied in the SIMMOD simulations using random seeds and probability distributions. We investigate the impacts of these uncertainties on ground delay by running fast-time simulations with peak-demand flight schedules at DTW. Through Monte Carlo simulations for each uncertainty factor, we quantify how the ground delay changes depending on the degree of uncertainty. We also repeat the identical stochastic simulations in respect to each uncertainty factor with both scheduled and optimized pushback times as inputs for comparison. The results confirm that surface traffic optimization based on a deterministic model performs reasonably even in the presence of certain types of uncertainties.

1.5 Organization of the thesis

The organization of this thesis is as follows. Chapter 2 proposes two optimization approaches for taxiway and runway scheduling. The first method is a unified model that simultaneously optimizes both runway and taxiway schedules. The alternative approach is to first find an optimal runway schedule, and then optimize the taxiway schedule. These two optimization architectures are evaluated with actual flight schedules at DTW. Fast-time simulations using SIMMOD are also implemented to assess the benefits of the gate-holding strategy used in both optimization approaches.

Case studies at DTW are described in Chapter 3 to analyze the effectiveness of the proposed optimization approaches. For several high density traffic scenarios, we optimize the initial flight schedules with respect to different optimization cases based on the proposed approaches, measure various airport performance metrics from the optimized flight schedule data, and compare them. Through comparisons between the scenarios, we also investigate the effects of aircraft fleet mix ratio and demand fluctuation in the flight schedule.

In Chapter 4, we compare two departure control approaches for efficient surface traffic management: individual aircraft trajectory-based control and aggregate

queue-based control. These control methods are implemented for high traffic demand scenarios and evaluated with respect to various airport performance metrics. We also consider gate conflicts between controlled departures and arrivals sharing the same gates and discuss possible solutions to mitigate gate conflicts.

Chapter 5 deals with the impact of uncertainty on airport performance. We develop a stochastic simulation model for uncertainty studies using SIMMOD. This model is used to test the effects of uncertain elements in airport operations such as pushback times, taxi speeds, and inter-departure times. We also investigate whether surface traffic optimization based on a deterministic model can still provide benefits in the presence of uncertainties. Chapter 6 concludes with a summary and extensions for future research.

Chapter 2

Optimization architectures for taxiway and runway scheduling

In this chapter, two different approaches to optimizing taxiway and runway schedules in airport operations are proposed and compared.

2.1 Modeling assumptions

A framework for modeling the real airport surface operations is generally constructed based on some assumptions. These assumptions are needed to simplify the complex situations in the real air traffic control environment with some level of reliability. Most of the assumptions have been established in other research in a similar manner [8,74].

The following describes the conditions that the proposed optimization models for airport runway and taxiway scheduling assume fundamentally. Note that the optimization models developed in this chapter are deterministic, but the uncertainty in airport operations will be considered in Chapter 5.

1. Airports have standard taxi routes in a given runway configuration. Therefore, given runway and gate, the taxi route of each flight is pre-defined.
2. Nominal taxi speed in free flow condition is given. Therefore, given the length of the taxiway, the minimum travel time on every taxiway link can be obtained.

When calculating the travel time on each link, the taxi speed values are assumed independent of aircraft types and weight classes.

3. The scheduled pushback times for departures and the estimated landing times for arrivals are given.
4. The preparation time for taxi-out is fixed and same for all flights. There is also no uncertainty in the pushback process. So, departures are pushed back as scheduled by the optimization model.
5. Airlines accept constrained position shifting (up to 2) in takeoff sequencing from the perspective of fairness.
6. Flights moving on the airport surface can meet the passage times at control points determined by optimization along taxi routes.

2.2 Integrated approach

The best way to integrate taxiway scheduling and runway scheduling and optimize them together is to put both objectives into a single optimization model. The single mixed-integer linear programming (MILP) model for taxiway and runway scheduling is proposed in this section. This model is basically obtained by modifying the MILP model for taxiway scheduling proposed by Rathinam et al [104].

2.2.1 Decision variables

For the aircraft taxi-scheduling problem, several MILP models have been proposed and improved by prior researchers, as described in the previous chapter [104,117,125]. These MILP models have two kinds of decision variables: 1) the continuous time variables for the passage times at nodes along the taxi routes of flights, and 2) the binary sequencing variables for determining the sequence of two flights at intersection nodes and runway thresholds where these flights may reach at the same or close time.

2.2.2 Objectives

For efficient taxiway scheduling, the model is designed to minimize the sum of taxi times of the flights moving on the ground within a given time window for optimization. In this objective function, the taxi times can be categorized by taxi-out times for departures and taxi-in times for arrivals. The model is also to minimize the runway delay for runway schedule optimization. The runway delay can be defined as the difference between the optimized takeoff time and the earliest possible takeoff time.

2.2.3 Constraints

The MILP model includes several important operational constraints which should be taken into account in airport operations. First of all, flights need to meet their schedules. Departing flights can leave their gates after the scheduled pushback times, by which passengers complete to board and crews are ready to depart. To minimize taxi-out time and save fuel burn, flights can be held at the gate for a while with engines off, depending on the congestion level on the surface. This is called “gate-holding strategy,” which can be utilized for schedule optimization of departure flights. However, the flights should leave the gate before the maximum gate-holding time because another arriving flight may want to use the same gate for disembarkation and unloading. Arrivals are assumed to land on the assigned runway at the estimated landing times, expressed as a fixed time constraint in the model. Also, flights moving on the taxiway need to obey taxi speed limitations under the airport operation rules.

More importantly, all the aircraft should keep the safety requirements. Taxiing aircraft have to keep some separation distance from the leading aircraft on the taxiway and ramp areas. Due to a similar reason, the following aircraft cannot overtake the leading one on the same taxiway. Also, the flights moving on the airport surface must not make any head-on conflicts at taxiway intersection points or on bi-directional taxiways. In order to avoid wake turbulence generated by the leading aircraft just after takeoff, the following aircraft on the departure runways should keep the separation requirements, which are dependent upon the weight classes of the consecutive flights.

2.2.4 Mathematical formulation

Incorporating the objective function and constraints described above, the mathematical formulation of the single MILP model for runway and taxiway scheduling can be expressed as follows.

$$\begin{aligned}
& \text{minimize } \sum_{i \in \mathcal{D}, r \in \mathcal{R}} \alpha_r (t_{i,r} - \text{EarliestOffT}_{i,r}) \\
& \quad + \alpha_d \left(\sum_{i \in \mathcal{D}, r \in \mathcal{R}} t_{i,r} - \sum_{i \in \mathcal{D}, g \in \mathcal{G}} t_{i,g} \right) + \alpha_a \left(\sum_{i \in \mathcal{A}, g \in \mathcal{G}} t_{i,g} - \sum_{i \in \mathcal{A}, r \in \mathcal{R}} t_{i,r} \right) \\
& \text{subject to } z_{ij}^u + z_{ji}^u = 1, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I} \tag{2.1} \\
& \quad t_{i,v} \geq t_{i,u} + \text{MinTaxiT}_{uv}, \forall i \in \mathcal{D} \cup \mathcal{A}, (u, v) \in \mathcal{E} \tag{2.2} \\
& \quad z_{ij}^u = z_{ij}^v, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u, v \in \mathcal{I}, (u, v) \in \mathcal{E} \tag{2.3} \\
& \quad z_{ij}^u + z_{ji}^v = 1, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u, v \in \mathcal{I}, (u, v) \in \mathcal{E} \tag{2.4} \\
& \quad t_{j,u} - t_{i,u} - (t_{i,v} - t_{i,u}) \frac{\text{Dsep}_{ij}}{l_{uv}} \geq -(1 - z_{ij}^u)M, \\
& \quad \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I}, (u, v) \in \mathcal{E} \tag{2.5} \\
& \quad t_{j,v} - t_{i,v} - (t_{j,u} - t_{j,u}) \frac{\text{Dsep}_{ij}}{l_{uv}} \geq -(1 - z_{ij}^v)M, \\
& \quad \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, v \in \mathcal{I}, (u, v) \in \mathcal{E} \tag{2.6} \\
& \quad t_{j,r} - t_{i,r} - \text{Rsep}_{ij} \geq -(1 - z_{ij}^r)M, \forall i, j \in \mathcal{D}, i \neq j, r \in \mathcal{R} \tag{2.7} \\
& \quad t_{i,r} \leq \text{EarliestOffT}_{i,r} + \text{MaxRunwayDelay}_{i,r}, \forall i \in \mathcal{D}, r \in \mathcal{R} \tag{2.8} \\
& \quad t_{i,g} \geq \text{OutT}_{i,g}, \forall i \in \mathcal{D}, g \in \mathcal{G} \tag{2.9} \\
& \quad t_{i,g} \leq \text{OutT}_{i,g} + \text{MaxGateHold}_{i,g}, \forall i \in \mathcal{D}, g \in \mathcal{G} \tag{2.10} \\
& \quad t_{i,r} = \text{OnT}_{i,r}, \forall i \in \mathcal{A}, r \in \mathcal{R} \tag{2.11} \\
& \quad t_{i,u} = \text{FrozenT}_{i,u}, \forall i \in \mathcal{D}' \cup \mathcal{A}', u \in \mathcal{N} \tag{2.12} \\
& \quad z_{ij}^u \in \{0, 1\}, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I} \tag{2.13} \\
& \quad t_{i,u} \geq 0, \forall i \in \mathcal{D} \cup \mathcal{A}, u \in \mathcal{N} \tag{2.14}
\end{aligned}$$

where \mathcal{D} and \mathcal{A} denotes departure set and arrival set, respectively. Similar deno-

tations represent \mathcal{I} for intersection node set, \mathcal{E} for taxiway link set connecting two nodes u and v , \mathcal{R} for runway set, and \mathcal{G} for gate set. M is a positive scalar with a large value. $t_{i,u}$ is the primary decision variable for the passage time of flight i at node u along its taxi route.

In the objective function, α_r is the coefficient for the runway delay in case that a flight takes off later than the earliest possible takeoff time. In addition, α_d and α_a are the coefficients of the total taxi-out time for departures and the total taxi-in time for arrivals, respectively.

Constraint (2.1) is the sequencing constraint to determine which flight goes first when two flights reach the same intersection node. Constraint (2.2) enforces the maximum taxi speed limit allowed at this airport and constrained by the aircraft performance, in terms of the minimum travel time on the taxiway segment. Constraint (2.3) makes two flights exiting a taxiway link maintain the same sequence as the order when entering the link. That is, this constraint prevents the following flight from overtaking the leading flight on the same taxiway. Constraint (2.4) is the sequencing constraint for bi-directional taxiway links, that avoids two flights entering the same taxiway link simultaneously and determines which flight enters the two-way taxiway link first. Constraints (2.5) and (2.6) describe the separation requirements between two flights taxiing at different speeds on the ground. Another constraint for safety is also included in (2.7) for runway operations. Since the required separation distance and time are dependent on the weight classes of the successive aircraft over the runway, “ $Rsep_{ij}$ ” can be different depending on the types of aircraft i and j . Time schedule constraints (2.8)-(2.11) define the latest takeoff time (EarliestOffT+MaxRunwayDelay) based on the earliest possible takeoff time and the maximum delay allowed for takeoff, the earliest gate-out time (OutT), the latest gate-out time (OutT+MaxGateHold) for departures, and the estimated landing-on time (OnT) for arrivals, respectively. Constraint (2.12) fixes the passage times of some flights that were already pushed back and are taxiing on the surface at the moment when optimizing the surface traffic. The passage times of these flights (FrozenT) come from the optimization result in the previous time window and are assumed to

be frozen so that cannot be updated at the current time window. Binary decision variable z_{ij}^u for sequencing between aircraft i and j at intersection node u is defined in (2.13). This variable is equal to one when aircraft i passes through the intersection point earlier than aircraft j , and equals to zero otherwise.

2.2.5 Strengths and weaknesses of the integrated approach

This model is fundamentally constructed based on the taxi scheduling model proposed by Rathinam et al. [104], but has been improved in several aspects. First, the single MILP model optimizes the runway schedule as well as the taxiway schedule by introducing an additional term for runway delay in the objective function. Without this term, the optimization would focus on minimizing the taxi time so that the takeoff time might be delayed further. That would also increase the gate-holding time excessively, which might aggravate gate utilization. Second, the model provides available takeoff time window having a reasonable range based on the earliest possible takeoff time, whereas Rathinam et al.'s model uses the scheduled takeoff time constraint. In their model, the constraint may make the solution infeasible sometimes if the scheduled time is too tight or inaccurate. Third, the model accounts for the existent flights taxiing on the surface that can interact with new flights in the current optimization period. This way, the model can be utilized for rolling horizon iterations as time progresses. Last, the model considers other safety constraints like collision avoidance in bi-directional taxiway.

The main strength of this single model is that we can implement runway scheduling and taxiway scheduling together. Using the single MILP model, we can simultaneously determine optimal departure sequence and takeoff times on runways, pushback times at gates for departures, and passage times of taxiing aircraft at control points along taxi routes, as well as predicted gate-in times for arrivals.

However, it sometimes takes a long time to find an optimal solution, especially at high traffic demand. Since the model is based on mixed-integer programming (MIP), the solver searches a number of branch-and-bound nodes to find the optimal solution, and needs to set up an appropriate MIP gap tolerance. Another problem is fairness

in the takeoff sequence among departure flights. To achieve the more efficient runway schedule and increase the runway throughput, the model allows excessive position shifting from the First-Come, First-Served (FCFS) sequence based on the original schedule. That would ruin the fairness of takeoff order between airlines, make the surface traffic more congested, and increase controllers' workload.

So as to overcome these problems, the other approach can be proposed. That is, instead of a single model, two separate optimization models for each purpose are used, but they are closely linked by sharing the same schedule data and operational conditions. In this approach, two optimization processes are sequentially implemented: runway scheduling first, and then taxiway scheduling.

2.3 Three-step approach

2.3.1 Methodology

The sequential process aiming at optimizing runway and taxiway schedules follows the three steps as described below.

Step 1 is to estimate the earliest possible takeoff times for departures. The earliest possible takeoff time of a departure flight can be computed by adding the unimpeded taxi-out time to the scheduled pushback time. The unimpeded taxi time is obtained based on the distance from gate to runway along the given taxi route and on the nominal taxi speed. The information about surface operations used in this step is same as the data used in Step 3 for taxiway scheduling so that the consistency on assumptions is maintained during the entire optimization process.

Next, Step 2 is to optimize the departure schedules at runways using a runway scheduling algorithm. At this step, the runway scheduling algorithm determines departure sequence and takeoff time schedule, considering the separation requirements over runways and the other conditions like available time window. The initial takeoff times used in optimizing runway schedules are assumed to be same as the earliest possible takeoff time from Step 1. This assumption makes the takeoff time window

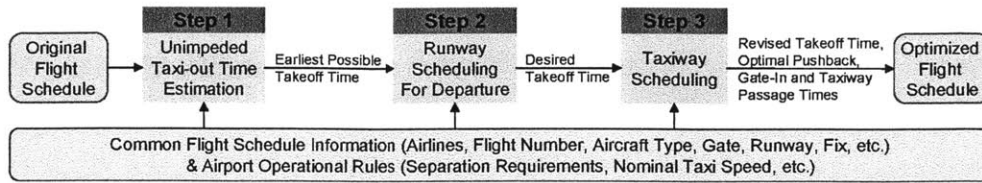


Figure 2-1: Sequential process diagram for three-step approach

in the next step reasonable.

Then, Step 3 is to optimize taxiway schedules using a MILP model. The MILP model determines optimal pushback times for departures, gate-in times for arrivals, and passage times at intersections on taxiways. For minimizing taxi-out times, the gate-holding strategy is applied. While optimizing aircraft taxi schedule and finalizing takeoff times of departures, both takeoff times from Step 1 and Step 2 are used. That is, the earliest possible takeoff time for a departure from Step 1 defines the lower bound of the available departure time window, and the optimized takeoff time from Step 2 is used as a guideline to determine the final takeoff time accounting for the taxiway conditions (e.g., potential conflicts with other aircraft).

This sequential process is illustrated in Figure 2-1. While taking these three steps, flight schedule information and airport operational rules are commonly shared. Details of each step are described as follows.

2.3.2 Step 1: Taxi-out time estimation for departures

The purpose of this step is to estimate the earliest possible takeoff times of departures, which will be a reference for optimizing the takeoff schedule and calculating the runway delay in the following steps. The estimated takeoff times are used in determining the initial takeoff sequence on the FCFS basis in Step 2. These times are also utilized to find the optimal takeoff times in Step 2 and Step 3, taking the separation requirements between consecutive takeoffs into account.

In this step, the unimpeded taxi-out time of each flight is first calculated based on the travel distance from its gate to the assigned runway along the given taxi route and on the nominal taxi speed. Some operational rules such as holding for crossing active

runways, taxiway and takeoff clearance procedures, and takeoff time from throttle-up to wheels-off are also considered. Adding this unimpeded taxi time to the scheduled pushback time provides the earliest possible takeoff time. However, the actual takeoff time can often be different from the earliest takeoff time of the flight because of other aircraft already arrived in departure queues. When a flight enters a departure queue, the flight may have to wait for a while until it uses the runway mainly due to the separation requirements.

2.3.3 Step 2: CPS algorithm for runway scheduling

For departure runway scheduling in Step 2, any algorithms introduced in Section 1.3.1 can be adopted. In this thesis, we will use the dynamic programming-based algorithm proposed by Balakrishnan et al. [13,14,15] because of its good computational performance and fairness in takeoff order.

Constrained Position Shifting (CPS)

This algorithm introduces Constrained Position Shifting (CPS), first proposed by Dear [40], for the fairness of the runway usage sequence in runway scheduling. Under the CPS method, the deviation from the FCFS takeoff sequence is limited while finding the optimal takeoff order. The restricted deviation from the FCFS order is denoted by the maximum number of position shifts, k , and the resultant scenario is referred to as a k -CPS case. The CPS concept helps not only maintain equity among aircraft operators, but also reduce the workload of controllers, by preventing a specific flight from waiting relatively for a long time before using the runway.

Other constraints

Besides the limited flexibility afforded to air traffic controllers, runway schedules are subject to several operational constraints. These constraints include the minimum separation requirements, available departure time windows, and precedence conditions between aircraft pairs.

The Federal Aviation Administration (FAA) regulates the minimum spacing be-

tween successive takeoffs to avoid the danger of wake turbulence. These separation requirements depend on the weight classes of the leading and trailing aircraft based on the maximum takeoff weight capacity [47]. The departure runway schedule also has to satisfy downstream separation requirements such as Miles-in-Trail (MIT) constraints at departure fixes. These metering constraints are imposed on the departures assigned to the same departure fix, which may not be consecutively operated at the runway.

The possible takeoff times of aircraft are also considered as constraints in runway scheduling. These constraints are basically in the form of time windows comprised of an earliest and a latest time of departure for the aircraft. The earliest takeoff time of a departing flight is obtained from the earliest possible takeoff time estimated in Step 1, whereas the latest takeoff time is determined by accounting for the acceptable levels of delay for the aircraft on the airport surface. These time windows can also be restricted by additional constraints such as the Departure Sequencing Program (DSP) and Expected Departure Clearance Times (EDCTs) used in Ground Delay Program (GDPs), and Approval Request (APREQ) procedures [32].

Lastly, there could be precedence constraints imposed on the departure sequence. These constraints state that aircraft i must take off before aircraft j in the algorithm. They can arise due to overtaking constraints on the ground movement, airline preferences from banking operations, or high priority flights.

Basic CPS framework

To solve the runway scheduling problem under CPS, a directed acyclic graph expressing every feasible takeoff sequence as a path in the network is first introduced. The scheduling problem is then solved using dynamic programming on this CPS network.

The CPS network consists of n stages, in addition to a source and a sink. Each stage corresponds to an aircraft position in the final sequence. A node in stage p of the network corresponds to a subsequence of aircraft of length $\min\{2k+1, p\}$, where k is the maximum position shift. For example, for $n = 5$ and $k = 1$, the nodes in stages 3, \dots , 5 represent all possible subsequences of length $2k+1 = 3$ ending at that stage,

$n=5, k=1$	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
Possible	1	1	2	3	4
last	2	2	3	4	5
aircraft		3	4	5	

Table 2.1: Possible aircraft assignments for $n=5, k=1$.

while the stage 1 contains a node for every possible sequence of length 1 starting at position 1 and the stage 2 contains a node for every possible sequence of length 2 ending at position 2. The network is generated by finding all possible sequence combinations of aircraft assignments to each position in the sequence (Table 2.1).

For each node in stage p , we draw directed arcs to all the nodes in stage $p + 1$ that can follow it. Figure 2-2 shows the network for $n = 5$ and $k = 1$. For example, the node (2 1 3) in stage 3 is a successor of node (2 1) in the previous stage (stage 2) and can precede the nodes (1 3 4) or (1 3 5) in the next stage (stage 4). The path (2)→(2 1)→(2 1 3)→(1 3 4)→(3 4 5) represents the aircraft sequence (2 1 3 4 5).

Some nodes that cannot belong to any path from source to sink are removed from the network. These nodes are shown in gray in Figure 2-2. By this process, we can produce a “pruned” network, which is significantly smaller than the original network. Precedence constraints may further reduce the size of the network.

Objective: Minimizing the sum of runway delays

The basic objective of the CPS algorithm used in the three-step approach is to minimize the sum of runway delays for departures, where the runway delay of a departing flight is the difference between the actual takeoff time and the earliest possible takeoff time. The CPS algorithm with this objective can be achieved using a modification of the algorithm to minimize the total landing cost of arrivals proposed in [83].

Given a set of departing aircraft, the *makespan* is defined as the takeoff time of the last aircraft, or in other words, the completion time of the takeoff sequence. As a first step, given a FCFS schedule, we determine a range of feasible makespan values. A trivial lower bound on the makespan is the minimum value among the earliest

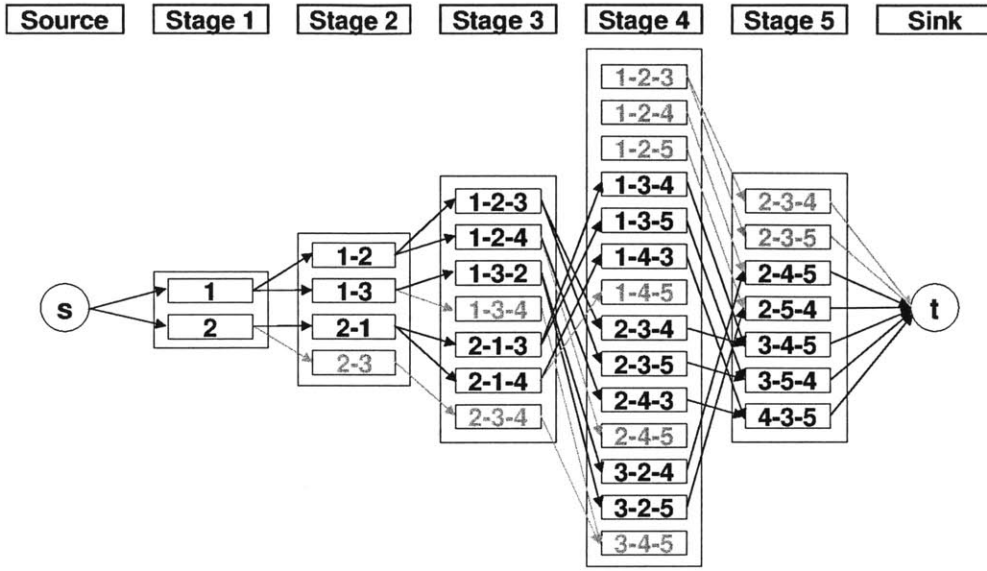


Figure 2-2: CPS network example for $n = 5$, $k = 1$

takeoff times of all aircraft that could take off last in the sequence. Similarly, the maximum value among the latest takeoff times of all aircraft that could take off last in the sequence would provide an upper bound on the makespan.

For each feasible value of the makespan, we consider all possible k -CPS sequences, and determine the optimal schedule that has the minimum total takeoff cost using a dynamic programming recursion. We first define the following variables:

- $\ell(x)$: The last aircraft of node x
- $\ell'(x)$: The second-from-last aircraft of node x
- $P(x)$: Set of nodes that are predecessors of x
- $\mathcal{I}(j)$: Set of times during which aircraft j could depart
- $c_j(t)$: Takeoff cost of departing aircraft j at time t
- t_j : Takeoff time of aircraft j
- e_j : Earliest possible takeoff time of aircraft j
- $\delta_{i,j}$: Minimum separation between aircraft i and j

Let $W_x(t_j)$ be the minimum value of the sum of takeoff costs that is accumulated until $l(x)$ takes off at time t_j . The objective of this algorithm is minimizing the total takeoff cost, that is equivalent to minimizing the sum of takeoff delays of all aircraft,

given the scheduled takeoff times.

For an arc (x, y) in the CPS network, the sum of takeoff costs from the first aircraft of the sequence to the last aircraft $\ell(x)$ of a node x , $W_x(t_{\ell(x)})$, is used to calculate the sum of takeoff costs from the first aircraft to the last aircraft $\ell(y)$ of the next node y , $W_y(t_{\ell(y)})$ using the following dynamic programming recursion:

$$W_y(t_{\ell(y)}) = \min_{x \in P(y)} \{W_x(t_{\ell(x)})\} + c_{\ell(y)}(t_{\ell(y)}), \quad (2.15)$$

$$\forall t_{\ell(y)} \in \mathcal{I}(\ell(y)) : t_{\ell(y)} - t_{\ell(x)} \geq \delta_{\ell(x), \ell(y)}$$

For a node y in the first stage, since there are no previous takeoff costs, the takeoff cost is given by $W_y(t_i) = c_i(t_i)$, where i is the last aircraft of the node. For example, i can be 1, 2, or 3, when the maximum number of position shifts allowed is equal to 2 ($k = 2$).

The dynamic programming recursion determines the total takeoff cost W for all nodes in stage n for all feasible time periods. The minimum cost schedule for a given makespan t is the minimum over all nodes x in stage n of $W_x(t_{\ell(x)})$, such that $t_{\ell(x)} = t$. Comparing $W_x(t_{\ell(x)})$ values for all nodes x in stage n , we can also determine the sequence and takeoff times of aircraft that minimizes the total takeoff cost of the schedule.

While minimizing the sum of takeoff delays, the takeoff cost of departing aircraft j at time t , $c_j(t)$, is equal to the takeoff delay from the earliest possible takeoff time of the aircraft ($c_j(t) = t_j - e_j$). However, this cost can be substituted by other values such as fuel burn, additional operating cost due to the delay, and a cost function with weighting factors incorporating airline preferences, according to the purpose of the algorithm.

Objective: Maximizing runway throughput

Another important objective is maximizing the throughput of the runway, which is equivalent to minimizing the makespan for a given set of aircraft (the static case). This objective can be used for the departure runway scheduling, instead of minimizing

the sum of takeoff delays. This can be obtained by using the same CPS framework with a simpler dynamic programming recursion, which is described as follows.

Based on the same CPS network and variables, we want to find the earliest time that the entire sequence can be completed, which is equal to the makespan. The values of $t_{\ell(\cdot)}$ can be computed by the following dynamic programming recursion. This recursion is solved using the boundary condition $t_{\ell(\cdot)} = e_{\ell(\cdot)}$ for all nodes in stage 1 [13,14].

$$t_{\ell'(y)} = \min_{x \in P(y)} t_{\ell(x)}; t_{\ell(y)} = \max \{ t_{\ell'(y)} + \delta_{\ell'(y), \ell(y)}, e_{\ell(y)} \} \quad (2.16)$$

For a fixed set of departures, the schedule with the minimum takeoff delay can be different from the schedule with the maximum runway throughput. For comparison, the evaluation of the optimal schedules from these two objectives will be performed in the next chapter with various traffic scenarios. The effects of the objective in runway scheduling on the taxiway schedule will also be analyzed. As a reference, more details about the CPS algorithm and its applications can be found in [13,14,15,82,83,84].

2.3.4 Step 3: MILP model for taxiway scheduling

The MILP model for taxiway scheduling used in Step 3 is similar to the single MILP model proposed in Section 2.2. The model has the same decision variables, which are the passage time of flight i at node u , $t_{i,u}$, and the binary sequencing variable between flights i and j at intersection node u , z_{ij}^u .

The objective of this MILP model is to minimize taxi-out times, taxi-in times, and the penalty for late takeoff. The penalty is applied by putting a large number into the coefficient, α_p , only if a flight departs later than the optimized takeoff time (DesiredOffT) from Step 2. Otherwise, the penalty is zero. Note that this setting allows earlier takeoff than the optimal takeoff time guided by Step 2 and gives flexibility in taxiway scheduling, depending on the taxiway conditions.

Basically, constraints are same as the single MILP model. The constraints account for the minimum travel time between nodes, the minimum separation on the surface

and over runways, no overtaking allowed along taxiways, conflict avoidance at intersection nodes and on two-way taxiways, and time schedules for pushback, takeoff, and landing. For the rolling horizon iterations, existing flights on the taxiway optimized at the previous iteration are also considered.

The basic mathematical formulation of the MILP model for taxiway scheduling in Step 3 is expressed as follows.

$$\text{minimize } \alpha_p \left(\sum_{i \in \mathcal{D}, r \in \mathcal{R}} \max[t_{i,r} - \text{DesiredOffT}_{i,r}, 0] \right)$$

$$+ \alpha_d \left(\sum_{i \in \mathcal{D}, r \in \mathcal{R}} t_{i,r} - \sum_{i \in \mathcal{D}, g \in \mathcal{G}} t_{i,g} \right) + \alpha_a \left(\sum_{i \in \mathcal{A}, g \in \mathcal{G}} t_{i,g} - \sum_{i \in \mathcal{A}, r \in \mathcal{R}} t_{i,r} \right)$$

$$\text{subject to } z_{ij}^u + z_{ji}^u = 1, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I} \quad (2.17)$$

$$t_{i,v} \geq t_{i,u} + \text{MinTaxiT}_{uv}, \forall i \in \mathcal{D} \cup \mathcal{A}, (u, v) \in \mathcal{E} \quad (2.18)$$

$$z_{ij}^u = z_{ij}^v, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u, v \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.19)$$

$$z_{ij}^u + z_{ji}^v = 1, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u, v \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.20)$$

$$t_{j,u} - t_{i,u} - (t_{i,v} - t_{i,u}) \frac{\text{Dsep}_{ij}}{l_{uv}} \geq -(1 - z_{ij}^u)M, \\ \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.21)$$

$$t_{j,v} - t_{i,v} - (t_{j,u} - t_{j,v}) \frac{\text{Dsep}_{ij}}{l_{uv}} \geq -(1 - z_{ij}^v)M, \\ \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, v \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.22)$$

$$t_{j,r} - t_{i,r} - \text{Rsep}_{ij} \geq -(1 - z_{ij}^r)M, \forall i, j \in \mathcal{D}, i \neq j, r \in \mathcal{R} \quad (2.23)$$

$$t_{i,r} \geq \text{EarliestOffT}_{i,r}, \forall i \in \mathcal{D}, r \in \mathcal{R} \quad (2.24)$$

$$t_{i,g} \geq \text{OutT}_{i,g}, \forall i \in \mathcal{D}, g \in \mathcal{G} \quad (2.25)$$

$$t_{i,g} \leq \text{OutT}_{i,g} + \text{MaxGateHold}_{i,g}, \forall i \in \mathcal{D}, g \in \mathcal{G} \quad (2.26)$$

$$t_{i,r} = \text{OnT}_{i,r}, \forall i \in \mathcal{A}, r \in \mathcal{R} \quad (2.27)$$

$$t_{i,u} = \text{FrozenT}_{i,u}, \forall i \in \mathcal{D}' \cup \mathcal{A}', u \in \mathcal{N} \quad (2.28)$$

$$z_{ij}^u \in \{0, 1\}, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I} \quad (2.29)$$

$$t_{i,u} \geq 0, \forall i \in \mathcal{D} \cup \mathcal{A}, u \in \mathcal{N} \quad (2.30)$$

The formulation above can be rewritten in a linear form by introducing a new decision variable for the penalty on late takeoff $y_{i,r}$ and adding some relevant ancillary constraints. The modified formulation is expressed as follows (See the change of the objective function and the new constraints (2.39) and (2.40)).

$$\text{minimize } \alpha_p \left(\sum_{i \in \mathcal{D}, r \in \mathcal{R}} y_{i,r} \right) + \alpha_d \left(\sum_{i \in \mathcal{D}, r \in \mathcal{R}} t_{i,r} - \sum_{i \in \mathcal{D}, g \in \mathcal{G}} t_{i,g} \right) + \alpha_a \left(\sum_{i \in \mathcal{A}, g \in \mathcal{G}} t_{i,g} - \sum_{i \in \mathcal{A}, r \in \mathcal{R}} t_{i,r} \right)$$

$$\text{subject to } z_{ij}^u + z_{ji}^u = 1, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I} \quad (2.31)$$

$$t_{i,v} \geq t_{i,u} + \text{MinTaxiT}_{uv}, \forall i \in \mathcal{D} \cup \mathcal{A}, (u, v) \in \mathcal{E} \quad (2.32)$$

$$z_{ij}^u = z_{ij}^v, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u, v \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.33)$$

$$z_{ij}^u + z_{ji}^v = 1, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u, v \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.34)$$

$$t_{j,u} - t_{i,u} - (t_{i,v} - t_{i,u}) \frac{\text{Dsep}_{ij}}{l_{uv}} \geq -(1 - z_{ij}^u)M,$$

$$\forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.35)$$

$$t_{j,v} - t_{i,v} - (t_{j,v} - t_{j,u}) \frac{\text{Dsep}_{ij}}{l_{uv}} \geq -(1 - z_{ij}^v)M,$$

$$\forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, v \in \mathcal{I}, (u, v) \in \mathcal{E} \quad (2.36)$$

$$t_{j,r} - t_{i,r} - \text{Rsep}_{ij} \geq -(1 - z_{ij}^r)M, \forall i, j \in \mathcal{D}, i \neq j, r \in \mathcal{R} \quad (2.37)$$

$$t_{i,r} \geq \text{EarliestOffT}_{i,r}, \forall i \in \mathcal{D}, r \in \mathcal{R} \quad (2.38)$$

$$y_{i,r} \geq t_{i,r} - \text{DesiredOffT}_{i,r}, \forall i \in \mathcal{D}, r \in \mathcal{R} \quad (2.39)$$

$$y_{i,r} \geq 0, \forall i \in \mathcal{D}, r \in \mathcal{R} \quad (2.40)$$

$$t_{i,g} \geq \text{OutT}_{i,g}, \forall i \in \mathcal{D}, g \in \mathcal{G} \quad (2.41)$$

$$t_{i,g} \leq \text{OutT}_{i,g} + \text{MaxGateHold}_{i,g}, \forall i \in \mathcal{D}, g \in \mathcal{G} \quad (2.42)$$

$$t_{i,r} = \text{OnT}_{i,r}, \forall i \in \mathcal{A}, r \in \mathcal{R} \quad (2.43)$$

$$t_{i,u} = \text{FrozenT}_{i,u}, \forall i \in \mathcal{D}' \cup \mathcal{A}', u \in \mathcal{N} \quad (2.44)$$

$$z_{ij}^u \in \{0, 1\}, \forall i, j \in \mathcal{D} \cup \mathcal{A}, i \neq j, u \in \mathcal{I} \quad (2.45)$$

$$t_{i,u} \geq 0, \forall i \in \mathcal{D} \cup \mathcal{A}, u \in \mathcal{N} \quad (2.46)$$

2.3.5 Expected benefits of the three-step approach

There are several benefits expected by this three-step approach. First, we can materialize efficient runway operations with various objectives such as maximum runway throughput, minimum takeoff delay, and minimum weighted sum of takeoff times. Since Step 2 is dedicated to optimal runway scheduling, various algorithms having different objectives can be applied for finding an optimal runway sequence and schedule in a reasonable time.

From equity point of view in takeoff order, the final sequence position of departures will not be largely deviated from the FCFS sequence based on the earliest possible takeoff time by virtue of the CPS method, although the departure sequence can be slightly adjusted in Step 3.

The taxiway schedule is also optimized while maintaining the separation requirements on runways. Using the gate-holding strategy, we can achieve less congested taxiway, lower taxi time, fewer stop-and-go situations, and less fuel burn during taxiing. We can expect that these benefits on the surface traffic may be almost same as the optimization results of the single MILP model, which will be shown with optimization results later.

Another merit is a fast solution time. It is expected to obtain an optimal solution of the MILP model in Step 3 quickly because runway scheduling, which makes more difficult to find an optimal solution of the single MILP model, is already almost done in Step 2. We will also compare the computation times of these two optimization approaches in order to verify this expectation.

2.4 Evaluation of two optimization approaches with one-day traffic data

In this section, the proposed approaches for runway and taxiway scheduling are applied to the actual flight schedule for evaluating their effectiveness and performance.

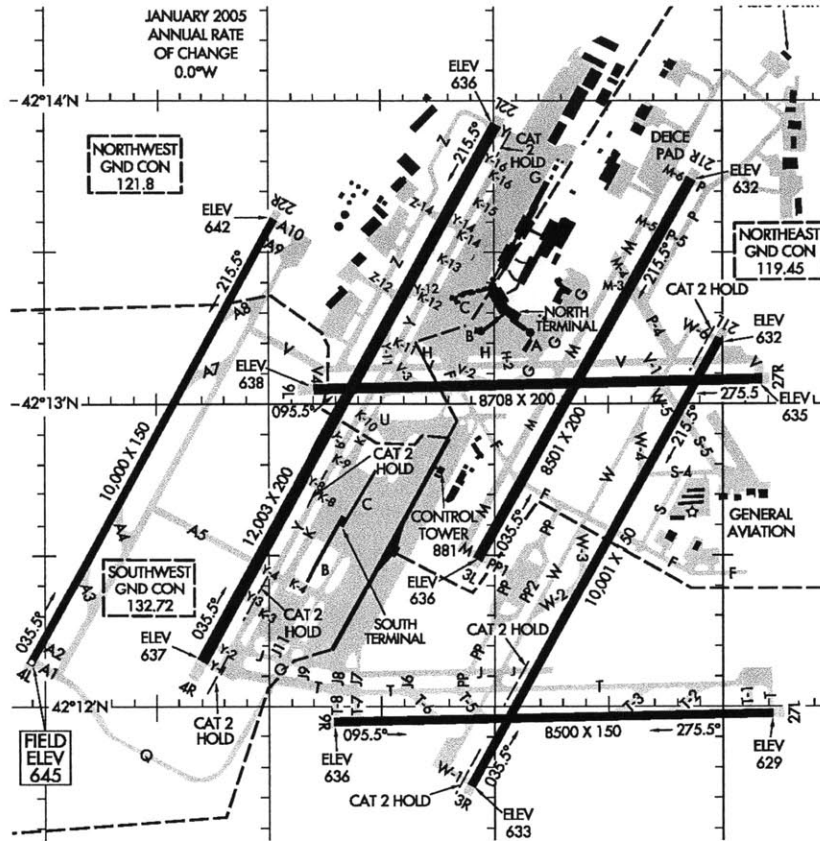


Figure 2-3: DTW airport layout (as of 2007)

2.4.1 Optimization set-up

The two optimization methods are tested and compared each other using the actual one-day flight schedule on 8/1/2007 at Detroit Metropolitan Wayne County airport (DTW). This date was a typical day of a busy summer travel season and in Visual Meteorological Conditions (VMC). On this date (except overnight hours between midnight and 5:45AM), a total of 1,294 flights were operated at this airport, including 656 departures and 638 arrivals. The runway configuration during the whole day was (22R, 27L | 21R, 22L), which is the most frequently used configuration. Figure 2-3 shows the airport layout at DTW.

Figure 2-4 illustrates the corresponding node-link network model that is used for the optimization models. The node-link model consists of 715 nodes that represent

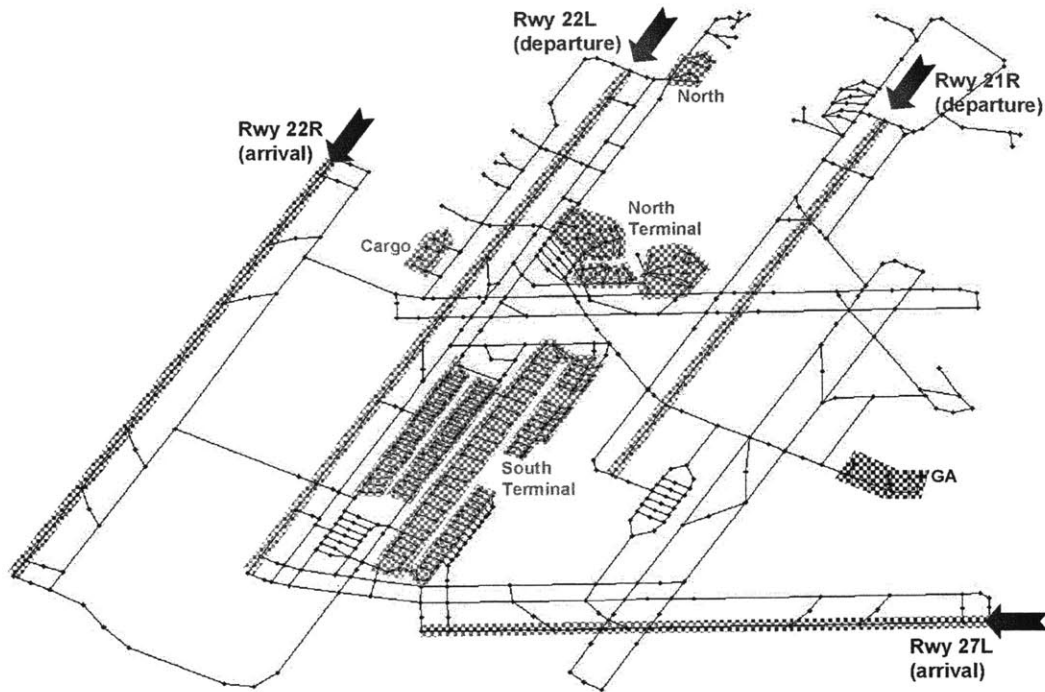


Figure 2-4: Node-link network model for DTW

significant control points on the airport surface and 863 links that connect adjacent nodes. The model contains 158 gates, including the parking areas for general aviation and cargo flights, as well as the main terminals in both North and South ramp areas. It is assumed that gates are enough to accommodate all the flights without duplicate gate assignment during peak periods based on the current traffic level. In the flight schedule, there are four aircraft types categorized by their maximum takeoff weights: Heavy, B757, Large, and Small. Note that heavy aircraft can depart off runway 22L only because the length of runway 21R is shorter than the minimum takeoff distance requirement for heavy aircraft.

For runway scheduling under CPS, three cases for takeoff sequencing are considered, depending on the maximum number of position shifts allowed: FCFS (no position shifts applied), 1-CPS, and 2-CPS. The objective of the scheduling algorithm used for this evaluation is to minimize runway delay (i.e., sum of takeoff times). The time window of the scheduling algorithm is 45 minutes, which starts at 5:45AM and

Leading Aircraft	Trailing Aircraft			
	Heavy	B757	Large	Small
Heavy	120	120	120	120
B757	90	90	90	90
Large	60	60	60	60
Small	60	60	60	60

Table 2.2: Minimum separation (in seconds) between takeoffs

moves forward by 15 minutes for the next iteration. This time horizon is discretized into small intervals of 5 seconds in length for fast computation. In the CPS algorithm, we consider the separation time requirements between successive departures shown in Table 2.2, depending on the weight classes of the leading and trailing aircraft.

The MILP models are coded in AMPL [50] and run by a CPLEX solver [70] on a 2.66GHz Dual core PC with 4GB RAM. The tolerance of optimization is set to ‘mipgap=0.0001’ and ‘integrality=1e-07’. The time limit of the solver is restricted up to one hour. The time window of the model is 30 minutes, moving by 15 minutes for the next iteration. In this way, the model can account for the frozen flights, which were already optimized in the previous iteration and are traveling on the taxiway in the current iteration. The time discretization used in these models is 5 seconds, which is small enough to control the flights moving on the surface.

In the objective function in the MILP models, the taxi time coefficients are commonly assumed as $\alpha_d = 1$ for departures and $\alpha_a = 2$ for arrivals because many airlines have operational procedures in which they do not turn all their engines on while taxiing out, thereby reducing fuel cost. In the integrated approach, the coefficient for runway delay α_r is set to 1. In the three-step approach, on the other hand, the coefficient for late takeoff α_p in the MILP model for taxiway scheduling is set to 100 as penalization for the flight not meeting the planned takeoff schedule and the following external cost to other flights. The minimum separation distance between taxiing aircraft on the ground ($Dsep_{ij}$) is assumed to be 150 meters in this evaluation, regardless of aircraft types, but can be varied, if needed. The minimum separation requirements between takeoffs ($Rsep_{ij}$) follow the same matrix in Table 2.2. In the optimization models, the maximum time that aircraft can be held at gate

Total time (in minutes)	CPS	MILP			
	Runway delay	Gatehold time	Runway delay	Taxi-out time	Taxi-in time
FCFS	378	380	423	5,876	6,192
1-CPS	371	367	410	5,876	6,192
2-CPS	365	375	417	5,876	6,191
FPS	-	393	419	5,859	6,192

Table 2.3: Optimization result for 1-day flight schedule

for $(\text{MaxGateHold}_{i,g})$ and the maximum runway delay allowed $(\text{MaxRunwayDelay}_{i,r})$ are set to 10 minutes and 15 minutes, respectively. Based on surface surveillance data from DTW, the nominal, free flow taxi speed values are assigned to 3, 7, and 18 knots on gate area, ramp area, and taxiways, respectively. The minimum taxi time on each link (MinTaxiT_{uv}) is calculated in advance using this taxi speed assumption and the length of each link.

2.4.2 Optimization result for one-day flight schedule

Table 2.3 summarizes the optimization results from different optimization approaches. In the first column, there are three cases from the three-step approach, depending on the maximum position shift (MPS) value. The ‘FPS’ case in the last row shows the optimization result of the single MILP model and represents Free Position Shifting, in contrast with the other constrained position shifting (k -CPS) cases. The second column shows the total amount of interim runway delay from the CPS algorithm in Step 2 of the three-step approach. The remaining four columns show total gate-holding time, sum of runway delays, total taxi-out time and total taxi-in time from the MILP models.

When we look at the second column in Table 2.3, we can find that the runway delay decreases as the k value (MPS) increases. After taxiway scheduling in Step 3, however, we can see that the runway delay increases by 52 minutes at most. This additional runway delay means that the takeoff sequence and the departure schedule are both affected a lot by taxiway conditions. For instance, the takeoff sequence may unintentionally change at intersection points while taxiing, and arrivals can have impacts on the movement of departing aircraft while either taxiing or crossing active

departure runways.

Also, excessive changes of the takeoff sequence in runway scheduling can lead to frequent interactions between flights and high congestion on the surface. While the 2-CPS case shows lower runway delay than the 1-CPS case after optimizing the takeoff order in Step 2, its final runway delay after taxiway scheduling becomes higher than the 1-CPS case. On the other hand, this expense of departures is compensated by the taxi-in time savings of arrivals, although the saving amount is small.

In the three-step approach, the total taxi-out and taxi-in times achieved by optimization are almost same, regardless of the maximum position shift allowed in the CPS algorithm, whereas the runway delay is dependent on the total gate-holding time applied to each case.

We now compare the optimization results from the three-step approach with the FPS case from the integrated approach. Compared to the 2-CPS case, it is shown that the total taxi-out time in the FPS case is reduced by holding some flights at gates longer, at the small expense of runway delay. When compared with the FCFS case, furthermore, the FPS case shows better runway delay and taxi times. It seems that these savings are mainly obtained by unlimited position shifting in the takeoff order. The optimal flight schedule in the FPS case experiences more frequent and further (up to 8) position shifting in the takeoff sequence over the FCFS order, which raises equity issue in sequencing.

Table 2.4 shows the computational performance of each optimization case. The sum of computation times in the table is for the whole day run having 73 iterations in total. The CPS algorithm for runway scheduling in Step 2 shows fast runtime for both FCFS and 1-CPS cases. However, the total runtime is dramatically increased when at most 2 position shifting is allowed. In the 2-CPS case, the average runtime for each iteration is about 29 seconds, which still shows good performance. The MILP model for taxiway scheduling in Step 3 of the three-step approach shows about 1 minute for total runs in all cases. It takes less than 1 second per iteration on average to optimize the flights within the given time window, that is amenable to practical implementation in the real world. On the other hand, the single MILP model for the

Runtime (in minutes)	CPS	MILP	
	Sum	Sum	Max
FCFS	0.10	0.79	0.07
1-CPS	1.27	0.82	0.12
2-CPS	34.82	1.12	0.35
FPS	-	166.03	64.08

Table 2.4: Total runtime for 1-day flight schedule

integrated approach has a runtime issue, as mentioned earlier. The FPS case takes 2 hours 46 minutes to run 73 iterations for the whole day flight schedule. Although most iterations take less than 2 minutes to find the optimal solutions, this optimization case includes 2 iterations resulting in sub-optimal feasible integer solutions with which the solver stops the optimization process due to the given time limit of 1 hour.

2.5 Departure planning method evaluation through fast-time simulation

In this section, we investigate how much benefits we can obtain from the surface traffic optimization over the current operations through fast-time simulation. For this evaluation, the whole day flight schedule at DTW used in the previous section is simulated.

To simulate the air traffic flow on the airport surface in the current operational conditions accurately, we use SIMMOD, which is a fast-time airport traffic simulation tool. SIMMOD can imitate the traffic flow of taxiing aircraft at a microscopic level, provide the travel time and delay on the surface for each flight, and visualize how the flight moves and interacts with other flights at the airport through a traffic flow animation.

For evaluating the benefits of the departure planning methods, the following procedure is performed. A baseline airfield model representing the Detroit airport layout is created in SIMMOD. Using this simulation model, we first run the air traffic simulation with the initial flight schedule in which departures are released at the scheduled pushback times without gate holding. Then, the pushback times determined by the

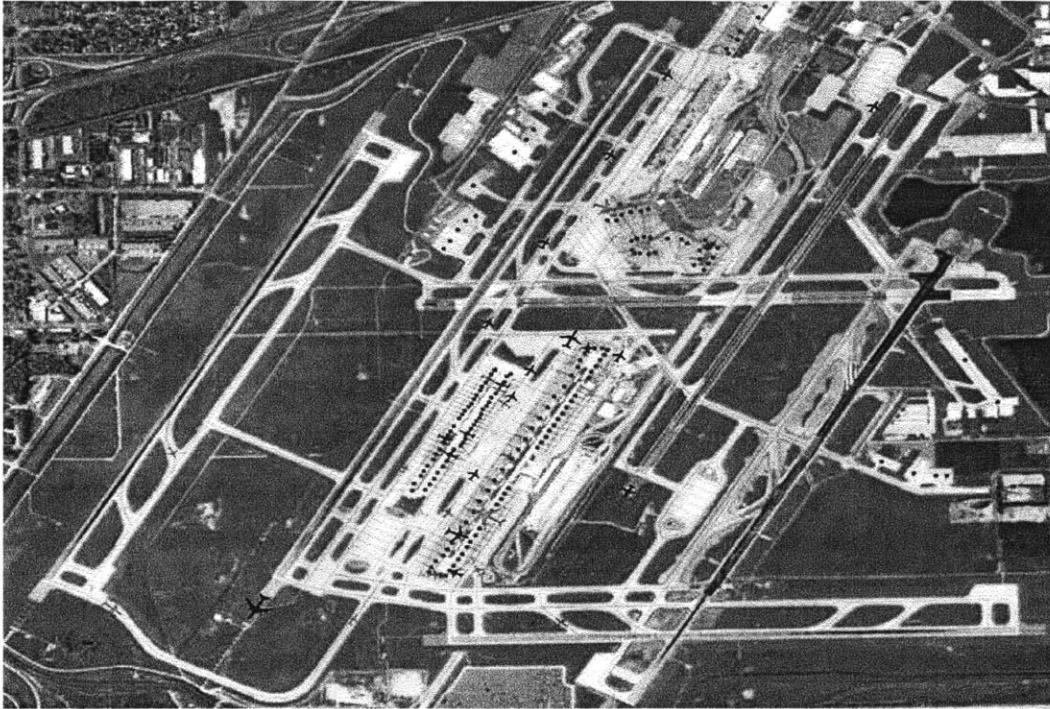


Figure 2-5: SIMMOD airfield model for DTW

proposed scheduling algorithms are entered as the simulation inputs, instead of the initial schedule. For a reasonable comparison, the common framework except the pushback times is used in the simulation. After implementing the simulations, the taxi times extracted from the simulation results are compared as a typical performance metric.

2.5.1 SIMMOD airfield model

The SIMMOD airfield model for DTW is based on the node-link network model in Figure 2-4, which was used for the optimization. SIMMOD can import the same coordinates of nodes and connectivity information of links in Google Earth KML format. The airfield and airspace model constructed in SIMMOD represents local airspace around the airport, airport ground surface, and gates in terminal buildings, as shown in Figure 2-5.

While constructing the SIMMOD airfield model, the information about airport

operations is shared with the optimization models. The same inputs with respect to airspace, airfield, and flight schedule are put into SIMMOD. The airspace information includes airport characteristics, details of airspace structures, and runway operational procedures such as separation distance, time interval, approach speed and runway occupancy time. The airfield input requires more detailed data on the surface such as runways, taxiways, gates, and departure queue area. Those data include taxiway operation conditions (link capacity, overtake rules, taxi speed, and directionality), gate operation rules (gate capacity, blocking state, and user airline list), and pre-defined taxi paths. The flight schedule data including airlines, flight number, aircraft type, origin/destination airport, airway route, runway, gate, and taxi path are recorded into the event file in SIMMOD. In the event file, pushback times for departures and landing times for arrivals come from the optimization results for the whole day flight schedule in Section 2.4.

2.5.2 Fast-time simulation result analysis

Figure 2-6 illustrates the total taxi times of departures from various cases. The first bar marked 'NoGH Simulation' represents the total taxi-out time in minutes from the SIMMOD simulation when the scheduled pushback times are applied. Since the gate-holding strategy is not used in this case, its taxi-out time is used as a reference to see the amount of taxi time savings in other optimization cases that control the pushback times of departures. The following bar pairs colored in green and purple show the total taxi-out times obtained from optimization and simulation, respectively. The dark bar over each taxi time bar represents the total gate-holding time for each case, so that the top indicates the relative total runway delay of departures.

In the bar graph, it can be easily seen that total taxi-out times (the sum of taxi-out times of all departures, excluding holding times at their gates) are significantly reduced by the gate-holding strategy. By focusing on the purple bars, we can directly compare the taxi times from different departure control methods in the same environment in SIMMOD. Compared to the 'No Gate Holding (NoGH)' case, it is observed that by holding departures at gates, the taxi time can be saved by 312 minutes in

total, which is about 5% of the total taxi-out time that departures of the day spent to travel on the ground. When the gate-holding strategy is applied, however, the total taxi-out times from simulation results have no significant difference, as in the optimization results.

We now draw a comparison between optimization and simulation in Figure 2-6. Compared to the optimization result, the corresponding simulation result shows a little higher taxi-out time, although the unimpeded taxi-out times are same. The gap between optimization and simulation mainly results from the feature of SIMMOD in which flights move on the taxiway link at a constant taxi speed unless possible conflicts are predicted. This is similar to the current surface operations, in the way that controllers cannot manage the speed of taxiing aircraft in details, whereas the optimization model actively interferes in it to achieve the optimal takeoff sequence. This characteristic in the simulation may dislocate the optimal departure schedule planned by the optimization, resulting in longer waiting times in the departure queues. More detailed analysis about this issue will be discussed later with various traffic scenarios in Chapter 3.

In Figure 2-6, the last two bars show the total taxi times from the optimization result of the single MILP model (FPS case) and from the corresponding simulation result, respectively. Compared to the three-step approach cases, the taxi-out time is slightly improved, but its total runway delay in the simulation is higher than the other cases because of the greater gap between optimization and simulation. This can be explained by the fact that excessive position changes in the takeoff order are made in the FPS case, resulting in more congestion on the surface.

For arrivals, the SIMMOD simulation results report the reduction of total taxi-in times by 0.8% in all cases, as shown in 2-7. The reason comes from the functional limitations of SIMMOD, which are no taxi speed control and no taxiway separation. The taxi-in time difference between optimization and simulation mostly happens in the ramp area near the terminal buildings, where arrivals interact with departures frequently. According to the SIMMOD outputs, the amount of the delay that arrivals experience is very small, which is about 0.1% over the total taxi-in time. It is also

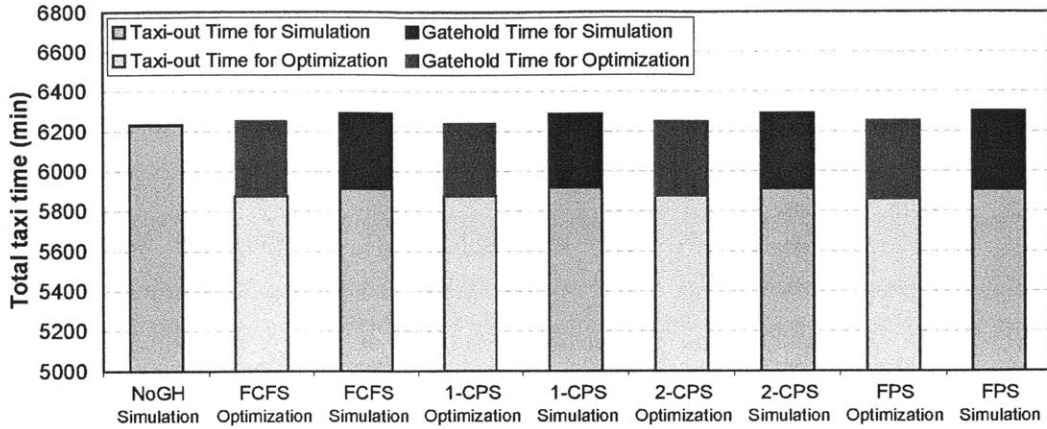


Figure 2-6: Total taxi-out time and gate-holding time for departures

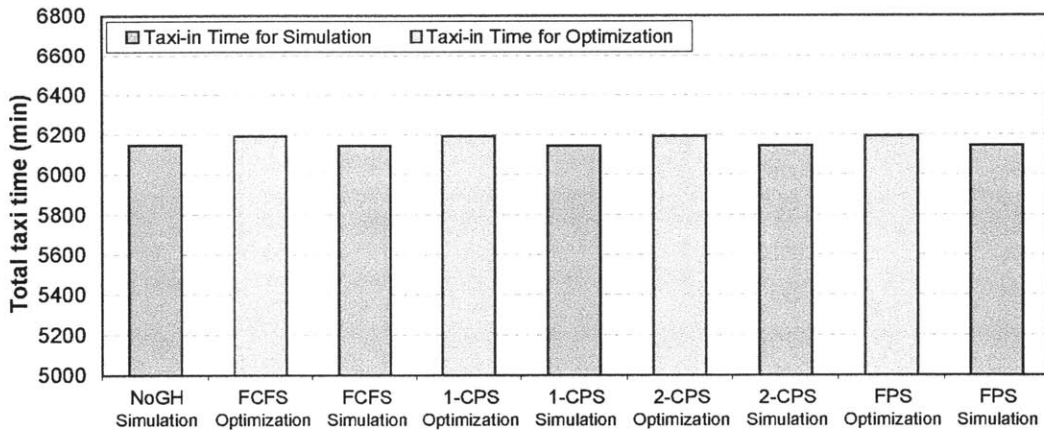


Figure 2-7: Total taxi-in time for arrivals

found that the most delay comes from runway crossings.

2.6 Conclusions

In this chapter, we proposed two approaches to optimizing runway and taxiway schedules. The first was the integrated approach based on the single MILP model. Another method was the three-step approach that sequentially combined two independent algorithms for runway scheduling and taxiway scheduling.

For evaluating their effectiveness, these approaches were applied to the actual

flight schedule at DTW with some assumptions involved in airport operations. The optimization results showed that both approaches provided significant taxi-out time savings, but the computational performance of the three-step approach was much better at the current level of traffic demand.

We also studied how much the taxi times could be reduced by the departure planning based on the surface traffic optimization, through fast-time simulations using SIMMOD. Compared to the simulation result based on the initial pushback schedule, the simulation results of the optimization cases using the gate-holding strategy showed significant taxi-out time savings.

Chapter 3

Case study at Detroit airport

3.1 Set-up for optimization and evaluation

The two optimization approaches proposed in the previous chapter will be investigated in more details with various traffic scenarios at Detroit airport (DTW) in this chapter.

First, we set up eight different optimization cases from the two optimization approaches, depending on the conditions in the scheduling algorithm. We also define the airport performance metrics that will be used to evaluate the optimization cases and compare them against each other. We then create high traffic demand scenarios at DTW for evaluation. In this chapter, three different traffic scenarios depending on fleet mix ratio and on demand patterns will be tested.

- Scenario 1. Constant high traffic demand with a fixed fleet mix ratio,
Heavy:B757:Large = 5%:10%:85%
- Scenario 2. Constant high traffic demand with a fixed fleet mix ratio,
Heavy:B757:Large = 10%:20%:70%
- Scenario 3. Cyclic high traffic demand with two peak times

For each scenario, the eight optimization cases are compared each other with respect to the various airport performance metrics. These optimization cases are also simulated in SIMMOD using their optimized flight schedules as inputs. The comparison between optimization and simulation results will show whether controlling

Case name	Optimization approach	Objective in departure runway scheduling	Position change limit in takeoff sequencing	Gate-holding strategy for departures	Objective in taxiway scheduling
FCFS ¹	Three-step approach	Not applied	0	Applied	Minimize total taxi-out /in times
1-CPSd		Minimize sum of runway delays	± 1		
2-CPSd			± 2		
1-CPSm		Minimize the last flight's takeoff time (<i>makespan</i>)	± 1		
2-CPSm			± 2		
FPSr	Integrated approach	Minimize runway delays	No limit		
FPSt	Not applied	No objective for runway scheduling			
NoGH				Not applied	

Table 3.1: Optimization cases for comparison

the pushback times only can achieve the same level of benefits as controlling the 4D trajectory of taxiing aircraft. The effects of the demand traffic characteristics such as fleet mix ratio and peak period are also analyzed through the result comparison between scenarios.

3.1.1 Optimization cases for comparison

For the case study at DTW, eight different optimization cases are defined. Table 3.1 summarizes those optimization cases that will be used in this chapter for evaluation. There are five cases derived from the three-step approach, depending on the runway scheduling objective and on the maximum position shift value. For the integrated approach, two cases are implemented. The first case named ‘FPSr’ is the original optimization model minimizing both taxi time and runway delay simultaneously, where the case name ‘FPS’ represents Free Position Shifting, in contrast with the other constrained position shifting (k -CPS) cases from the three-step approach. The FPSt case is a control group that minimizes the taxi time only by putting $\alpha_r = 0$ in the objective function. This case was added to see the impact of the runway delay term in the MILP model. Finally, the NoGH case presents the optimal taxiway and runway schedule from the same framework when no gate-holding is applied and departures leave their gates at their scheduled times. This case is used as a baseline to evaluate the benefits of gate-holding.

Category	Metrics
Gate	Average gate-holding time
Taxiway	Average taxi-out time
	Average taxi-in time
	Number of taxiing aircraft
	Number of stops
Runway	Average takeoff delay
	Largest takeoff delay
	Runway throughput
	Takeoff sequence changes
Interaction between taxiway and runway	Takeoff rate as a function of departure demand
Computational performance	Total runtime

Table 3.2: Airport performance metrics

These optimization cases will be implemented for three traffic scenarios in this chapter and compared each other using various airport performance metrics for each traffic scenario.

3.1.2 Airport performance metrics

Table 3.2 shows the various metrics that can be used for measuring airport performance in the case studies at DTW. These metrics can be categorized according to the resource in airport operations such as gate, taxiway, and runway.

At gates, the average gate-holding time per departure is measured. The gate-holding time is a difference between the initial pushback time given by a flight schedule and the controlled pushback time by optimization. If spots² are used as start points in the optimization model instead of gates, the waiting time at a spot becomes the gate-holding time. This duration can be calculated by subtracting the estimated spot arrival time of a departure from the time to enter the taxiway from the spot [74]. By definition, the gate-holding time is always zero in the NoGH case. As described in the formulation in Chapter 2, the maximum gate-holding time is limited as a constraint in the MILP model.

¹FCFS: First-Come, First-Served, CPS: Constrained Position Shifting, FPS: Free Position Shifting, NoGH: No Gate-Holding for departures (runway and taxiway schedule optimization to resolve conflicts only)

²“Spot” is the hand-off point between the airline ramp control and Tower control, marked on the pavement with a number.

On taxiway areas, typical performance metrics are the average taxi-out time for departures and the average taxi-in time for arrivals. The taxi-out time of a departure is defined as the difference between actual gate-out time and actual wheels-off time. Similarly, the taxi-in time of an arrival is actual gate-in time minus actual wheels-on time. In addition to the taxi times, the number of aircraft taxiing out can be counted to measure the level of taxiway congestion. The number of stops is also a good indicator to show how often the taxiing aircraft interact with other vehicles at significant locations on the ground such as taxiway intersections, runway crossings and departure queues. This metric can be used to better estimate the fuel burn and gas emissions on the ground [77].

For runways, runway throughput and the average takeoff delay are major performance metrics. The largest takeoff delay can be observed for fairness among departures. Takeoff sequence changes between initial estimates and final optimization results are also counted since this metric may be related to the workload of local controllers. In most prior researches, departure runway throughput (or takeoff rate) is expressed as a function of the number of aircraft taxiing out in order to see the interaction between runway and taxiway [113,115,116]. This relationship will be plotted in Section 3.4.2 for Scenario 3 which has variations in runway throughput.

To analyze the computational performance of the scheduling algorithms, total runtime taken to find the optimal solutions is measured. This metric is important because the scheduling algorithm should be able to support the controller's decision making in real time. If a time limit is set in finding a solution, the time limit frequency representing how often the solver reaches the time limit during successive iterations can be of interest.

In this chapter, the eight different optimization cases will be compared using the basic airport performance metrics among the items listed in Table 3.2, including taxi-out times, taxi-in times, takeoff delays, the number of position changes in takeoff sequencing, and total runtimes, for the constant high traffic demand scenarios. In Scenario 3, more various performance metrics will be investigated for the realistic flight schedules having peak times. These extended metrics include runway through-

put and the number of taxiing aircraft, which can be varied by cyclic fluctuations in demand.

3.2 Scenario 1: Constant high traffic demand with a fixed fleet mix ratio (5% heavy aircraft)

3.2.1 Traffic data for Scenario 1

To model a high traffic demand scenario at DTW, we first assume that flights are consistently supplied to this airport for 3 hours at the rate of 160 flights per hour, with 80 departures and 80 arrivals. This rate is twice the average hourly traffic demand at DTW in 2007 and close to its declared capacity, namely, 184-189 operations/h in optimal conditions and 168-173 operations/h in marginal conditions [48].

The detailed flight schedule data for individual flights, including scheduled push-back or landing times, gates and runways, are randomly generated by SIMMOD. Consistent with actual traffic data at DTW, the fleet mix ratio is assumed to be 5%, 10%, and 85% of heavy aircraft, B757, and large aircraft, respectively. As two runways are usually used for departures, runways are assumed to be balanced. For this experiment, 28 sets of flight schedule scenarios are generated and optimized using the eight different optimization cases.

3.2.2 Optimization results

Figure 3-1 shows the average gate-holding time and the taxi-out time per departure for each optimization case. The whiskers denote the standard deviation of the sum of the two times, across the 28 flight schedules. All the optimization cases from two different approaches show similar taxi-out times, except for the FPSt case that minimizes only taxi times, but at the expense of long pushback delays. The figure shows that the taxi-out time can be reduced by about 64 s per departure relative to the NoGH case through gate-holding. The takeoff times (sum of gatehold and taxi-out times) are similar, meaning that the gate-holding time translates to taxi-out time

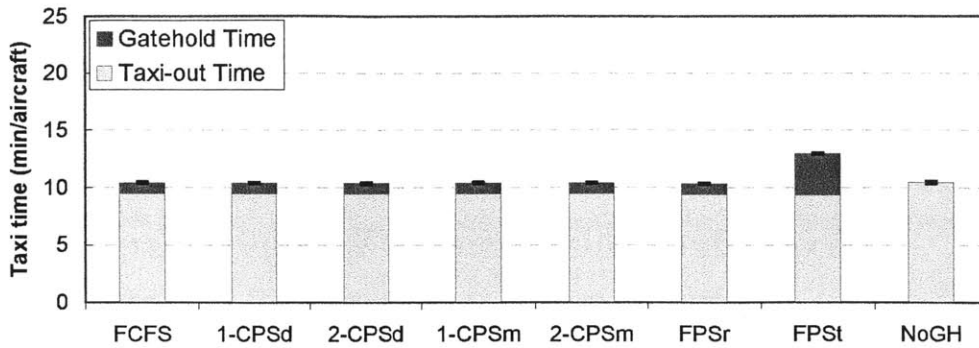


Figure 3-1: Scenario 1: Average taxi-out time per aircraft

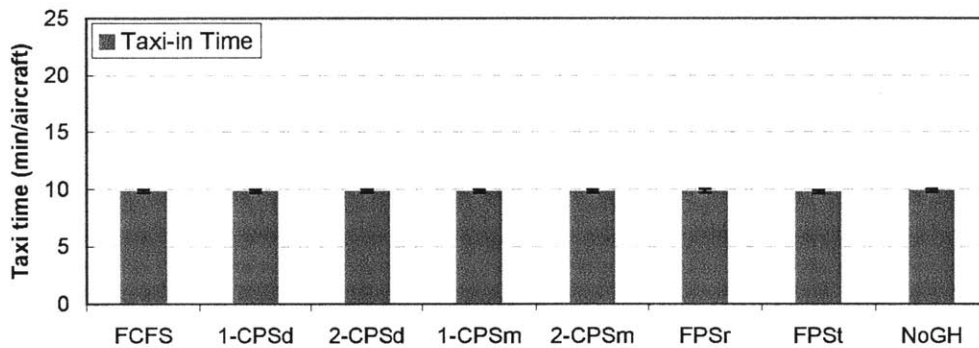


Figure 3-2: Scenario 1: Average taxi-in time per aircraft

savings. This result is same as the benefit analysis of the departure control strategy from the field tests at BOS [116].

The average taxi-in times shown in Figure 3-2 are also similar to one another (except for the FPSt case, which has a lower average taxi-in time at the cost of takeoff delays). It appears that holding departures at their gates has little effect on the arrivals.

Figure 3-3 shows the takeoff delay per departure for each of the eight optimization cases. The takeoff delay is defined as the actual takeoff time minus the earliest possible takeoff time (obtained by adding the unimpeded taxi time to the originally scheduled pushback time). In the three-step approach (left five cases in the graph), the takeoff delay from runway scheduling in Step 2 arises from the separation requirements between takeoffs. In Step 3, small additional delay occurs due to taxiway interactions,

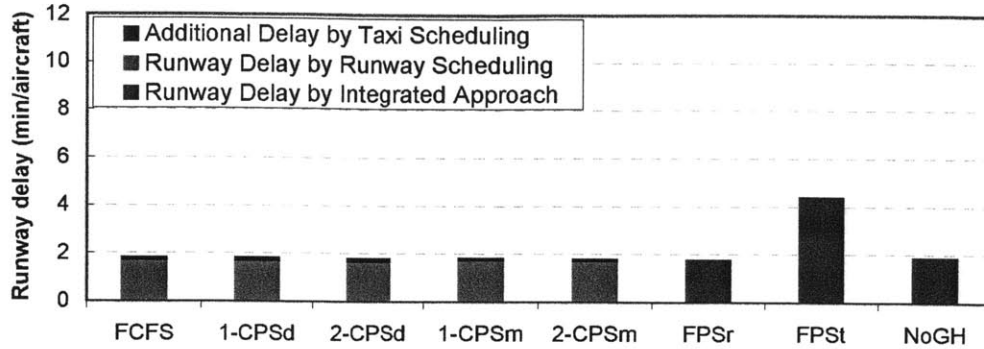


Figure 3-3: Scenario 1: Average takeoff delay per departure

leading to a little longer delay than the FPSr case in the integrated approach.

Compared with other optimization cases, the FPSt case shows a significantly larger runway delay because it does not consider runway delay (difference between actual takeoff time and earliest takeoff time) in the optimization. The runway delay term in the objective function is ignored in the MILP model for the integrated approach by making its coefficient zero, $\alpha_r = 0$. Without this runway delay term, the FPSt case tends to postpone the takeoff times of departures as long as possible by holding them at gates in order to minimize the total taxi time alone. However, it appears that we cannot obtain an additional reduction in the taxi-out times in this case, as shown in 3-1. The other optimization cases achieve taxi-out times close to the unimpeded taxi-out times, despite optimizing the runway schedule as well.

Some flights may not meet the initially assigned takeoff slots due to unexpected interactions on the taxiways. This could increase the workload of air traffic controllers. Figure 3-4 illustrates the number of position changes from the initial takeoff sequence relative to the earliest possible takeoff times for each departure runway. In the FCFS case, the takeoff sequence is determined on the first-come, first-served basis in Step 2, but about 10% of departures cannot follow the given sequence because of the interactions with other aircraft while taxiing. When the pushback times of departures are fixed, the impacts of taxiway interactions on the takeoff orders become larger. The NoGH case shows that more than 39% of departures experience at least one position shift in the takeoff sequence for both runways. In addition, a comparison of the FCFS,

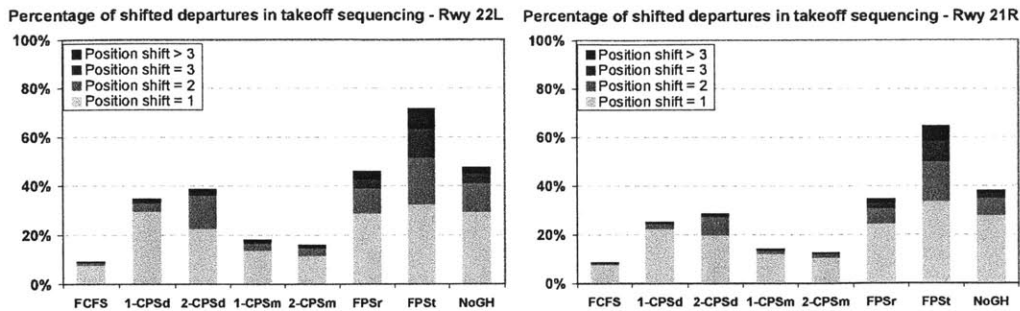


Figure 3-4: Scenario 1: Takeoff order changes from initially estimated sequence

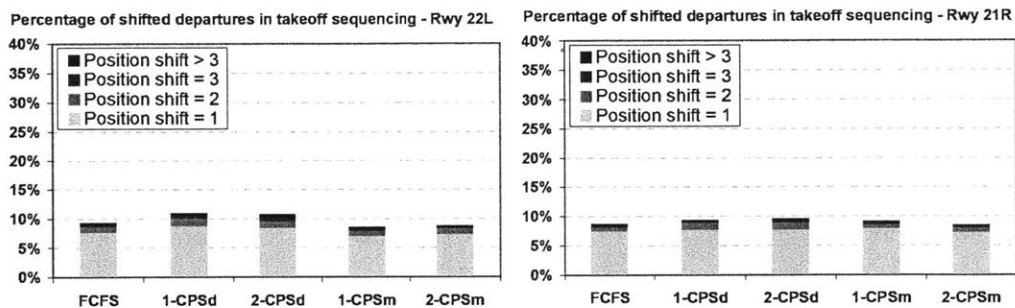


Figure 3-5: Scenario 1: Takeoff order changes between Step 2 and Step 3

1-CPSd, 2-CPSd, and FPSr cases demonstrates that the number of takeoff sequence changes increases as the position change limit increases. The integrated approach (i.e., FPSr and FPSt cases) shows more sequence changes than the other approach because the limit of the takeoff slot shifts is not constrained.

The effects of taxiway scheduling on the takeoff sequence can also be studied by observing the difference between the takeoff orders from Step 2 and Step 3 in the three-step approach, as shown in Figure 3-5 for each runway. In this scenario, 8-11% of flights cannot meet the optimal takeoff slots determined by the CPS algorithm. These position changes are a consequence of taxiway scheduling, and may result in additional delays to takeoffs.

Figure 3-6 compares the total runtimes of the different optimization cases for a 3-hour flight schedule period. The CPS algorithm used in the three-step approach is typically fast, except in the 2-CPSd case which takes 20 min to optimize the 3-hour long flight schedule. The MILP model used in Step 3 also shows good computational

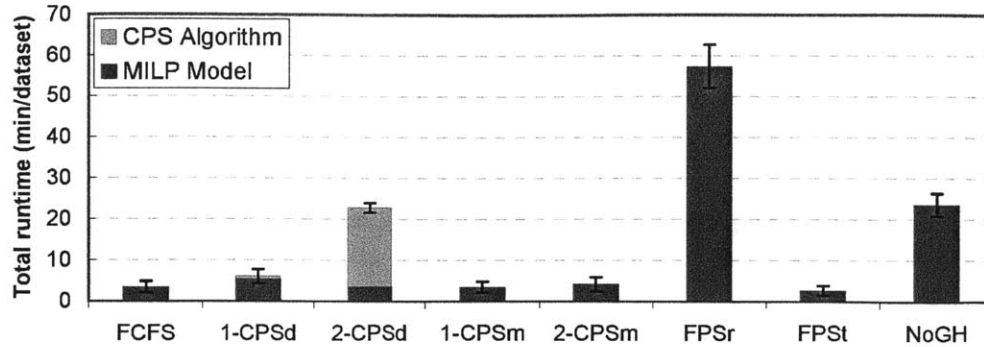


Figure 3-6: Scenario 1: Total runtimes

performance. As expected, the FPSr case for the integrated approach takes a long time to optimize the Required Times of Arrival (RTAs) of the taxiing aircraft, because it simultaneously implements both runway and taxiway scheduling. Considering that the resultant runtime for a 3-hour flight schedule includes 12 iterations and that every iteration the flights within each 30-minute time window are optimized, the average runtime per iteration of the FPSr case is less than 5 minutes. On the contrary, The FPSt case finds the optimal solution very fast because it does not account for the runway schedule and focuses on minimizing taxi times only. Lastly, the NoGH case needs a significant runtime because it determines the optimal takeoff times and RTAs of flights like the FPSr case, although the pushback times of departures are fixed.

3.2.3 Comparison with simulation results

Using fast-time simulations in SIMMOD, we can investigate whether the optimization strategies are valid in the current operational environment. Figure 3-7 illustrates the average taxi-out times per departure from both optimization and simulation for the eight different optimization cases. As described in Chapter 1 and 2, the optimization and the fast-time simulation correspond to the RTA control utilizing the advanced surface traffic control technologies and the pushback time control in the current operational conditions, respectively. According to the simulation results, the significant taxi-out time savings can be obtained by controlling pushback times only, compared to the NoGH case.

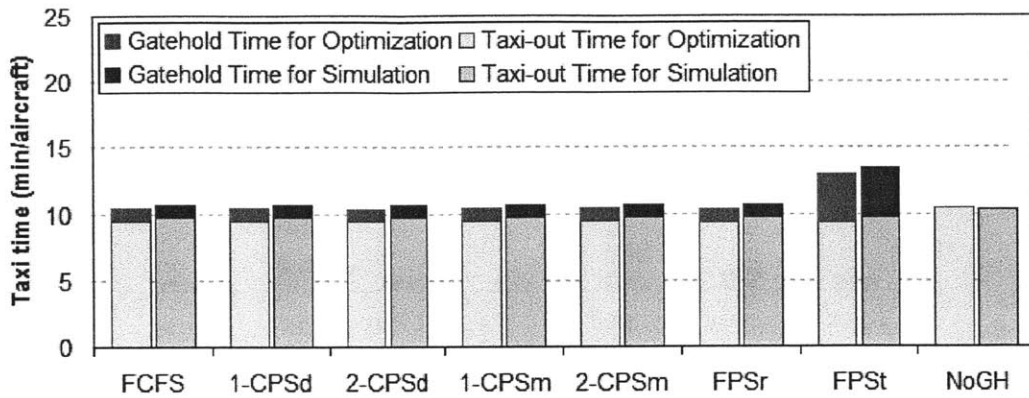


Figure 3-7: Scenario 1: Average taxi-out time comparison between optimization and simulation

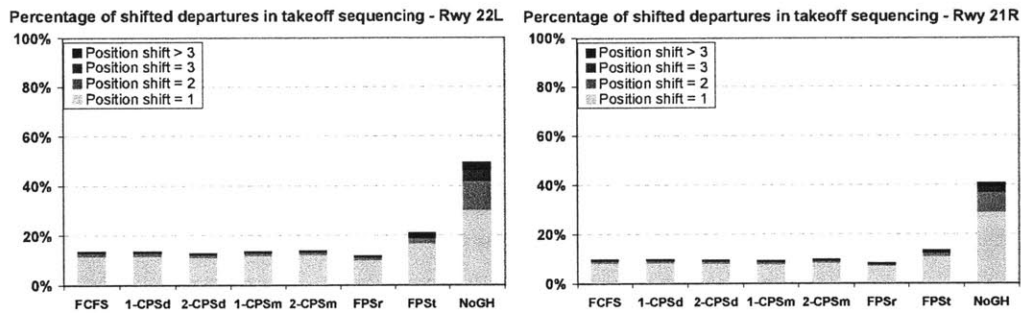


Figure 3-8: Scenario 1: Takeoff order changes between optimization and simulation

A comparison of optimization and simulation results for the FCFS (or the FPSr) case also shows that the RTA control can further decrease the taxi-out time by up to 16 s/aircraft. When the pushback times are only controlled, some departures interact with other departures or arrivals on the taxiway. These interactions increase waiting time in the departure queue since the takeoff sequence may change, as can be verified by observing the takeoff position changes between the optimized solution and the SIMMOD simulation. Figure 3-8 shows that some departing flights are affected by the RTA control at significant points on the taxiway. In addition, the high percentage of shifted departures in the NoGH case implicates that more holds at taxiway intersections are required when gate-holding is not applied.

For arrivals, the average taxi-in times from optimization and simulation are illustrated in Figure 3-9. The graph shows that the average taxi-in times from simulation

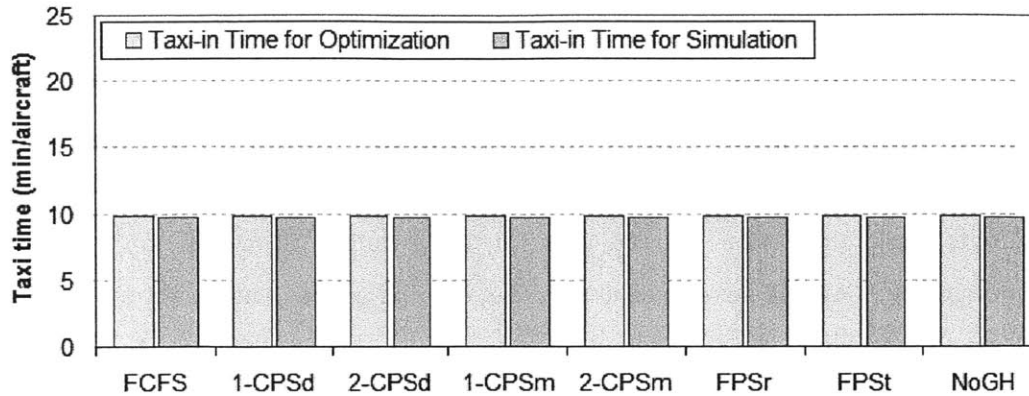


Figure 3-9: Scenario 1: Average taxi-in time comparison between optimization and simulation

are 9-10 seconds lower than the values from optimization in all the optimization cases. This difference stems from the functional limitations of the fast-time simulation tool. First, holding on the taxiways, which is frequently used in the RTA control for sequencing at intersections, is ignored in SIMMOD. Second, the taxiway separation is less restricted in SIMMOD, whereas the optimization model keeps the minimum separation distance between taxiing aircraft. In the optimization model, the constraints for the taxiway separation let arrivals slow down at some control points when the arrivals interact with departures on the ramp area or other arrivals in the taxi speed transition area. These reasons may lead the arrivals to reach their gates faster in simulation results, but the difference is not significant.

3.2.4 Effect of the objective in runway scheduling

In the three-step approach used for this scenario, we have tested two typical objectives in runway scheduling: minimizing the sum of runway delays and maximizing runway throughput. As mentioned earlier, it is noted that minimizing the sum of runway delays is the same as minimizing the average takeoff delay of departures and that maximizing the runway throughput is equivalent to minimizing the makespan of the given departure flights.

In most cases, runway scheduling algorithms provide benefits on both the runway

throughput and the average takeoff delay, regardless of its objective [60,83]. That is, the optimal takeoff schedule to minimize the average delay improves the throughput as well. Similarly, the maximum throughput schedule usually shows the better runway delay, compared to the FCFS takeoff schedule. However, the prior research work using Monte Carlo simulations showed that when the runway delay was minimized, the probability of having adverse effects on the other objective (runway throughput) was lowered than when the throughput was maximized. On the contrary, the maximum throughput solution sometimes showed a large deviation from the optimal runway delay.

When incorporating the runway scheduling with taxiway scheduling sequentially as in the three-step approach, the comparison result with regard to the two runway scheduling objectives can change because of the taxiway conditions. When optimizing the runway schedule, the earliest takeoff time of each flight is generally estimated based on the unimpeded taxi time, but the available takeoff time window may be shifted due to the interactions with other taxiing aircraft on the surface. This shift can sometimes make the optimal takeoff time solution suboptimal or infeasible in the worst case. The additional runway delay by taxiway scheduling shown in Figure 3-3 justifies the effect of taxiway situations.

The comparison demonstrates that the difference between runway scheduling objectives is narrowed by taxiway scheduling. In Figure 3-3, the 1-CPSd case shows a lower runway delay than the 1-CPSm case right after runway scheduling in Step 2, but their final runway delays after taxiway scheduling are almost same. In addition to the runway delay, the average taxi-out and taxi-in times achieved by optimization have no difference between the two objectives in runway scheduling, as seen by the comparison of the 1-CPSd, 2-CPSd, 1-CPSm, and 2-CPSm cases in Figures 3-1 and 3-2. It also seems that the maximum position shift (MPS) value in the CPS algorithm does not influence the taxi time performance.

Provided that the resultant system performance is the same, minimizing the makespan would be a better choice from the controller's point of view. According to Figure 3-4, the minimum makespan schedule requires small deviation from the

initially estimated takeoff sequence, whereas the takeoff order changes are doubled when the runway delay is minimized in Step 2 for runway scheduling. Furthermore, the computational performance shown in Figure 3-6 is also better in the minimum makespan solution (the 1-CPSm and 2-CPSm cases), which is critical for implementing the algorithm in a decision support tool.

The conclusion about the effects of the runway scheduling objective on the airport performance discussed in this subsection also seems valid for other air traffic scenarios having different demand characteristics such as fleet mix ratio and peak periods that will be described in the following sections.

3.3 Scenario 2: Constant high traffic demand with a fixed fleet mix ratio (10% heavy aircraft)

3.3.1 Traffic data for Scenario 2

Scenario 2 was designed for investigating the effects of aircraft fleet mix. The flight schedules in Scenario 2 were created in the same way as in Scenario 1, except for the fleet mix ratios, which was set to 10%, 20%, and 70% for heavy aircraft, B757, and large aircraft, respectively. In this scenario, the portion of heavy and B757 aircraft was doubled from Scenario 1, resulting in more heterogeneous fleet mix ratio. The detailed flight schedule data for each flight were generated by SIMMOD, as before. The same traffic demand and runway balancing were used. For this experiment, 28 different sets of flight schedules were generated and optimized in eight optimization cases.

3.3.2 Optimization results

Figures 3-10 and 3-11 illustrate the average taxi-out times and the average taxi-in times in the eight different optimization cases. A comparison of Figures 3-1 and 3-2 shows that the average taxi-out and taxi-in times in Scenario 2 are almost the same as in Scenario 1, when departures are controlled at their gates. In the NoGH case, by

contrast, the average taxi-out time increases by 20 s/aircraft compared to Scenario 1 with the increased portion of heavy and B757 aircraft in the fleet mix.

The metrics mainly affected by the fleet mix change are the gate-holding time, takeoff delay, and the takeoff sequence. Generally, increasing the proportion of heavy aircraft in the fleet mix ratio of flights reduces the runway capacity and increases the average waiting time for the next flight to use the runway. The increased separation time for takeoffs translates to the longer gate-holding time in the optimization results in order to reduce the taxi time, as seen in Figure 3-10. By comparing Figure 3-12 with Figure 3-3, we can see that the more heterogeneous fleet mix ratio leads to increased runway delay due to the separation requirements.

The number of takeoff order changes also increases by about 10% for all the cases, excluding the FCFS case, as shown in 3-13. This result mainly stems from the efforts to find the optimal takeoff sequence in runway scheduling. That can be verified by observing Figure 3-14, showing that the percentage of takeoff sequence changes by taxiway scheduling in Scenario 2 are similar to the percentage in Scenario 1.

Regardless of the fleet mix ratio in the flight schedule scenarios, the computational performances are similar. The total runtimes shown in Figure 3-15 have the same trend as the runtime results for Scenario 1, though there are variations case by case.

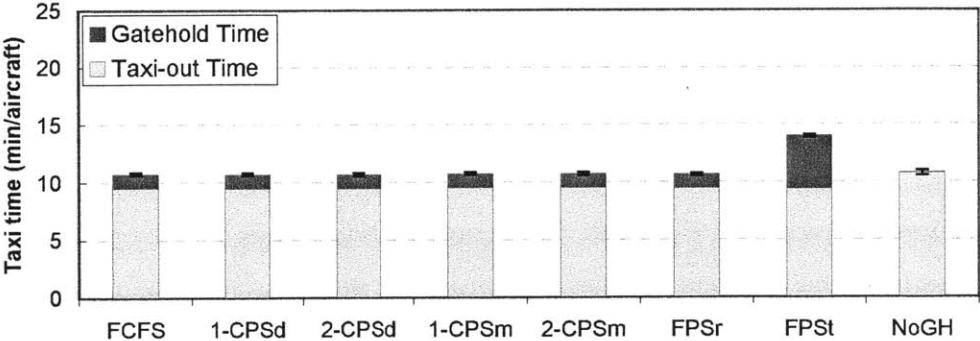


Figure 3-10: Scenario 2: Average taxi-out time per aircraft

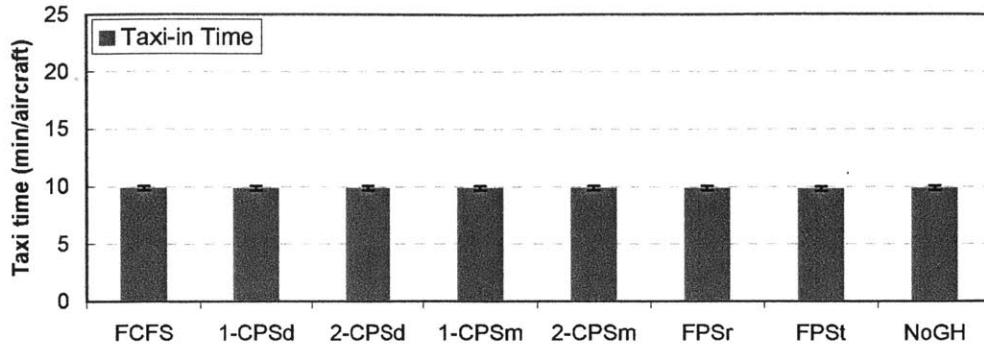


Figure 3-11: Scenario 2: Average taxi-in time per aircraft

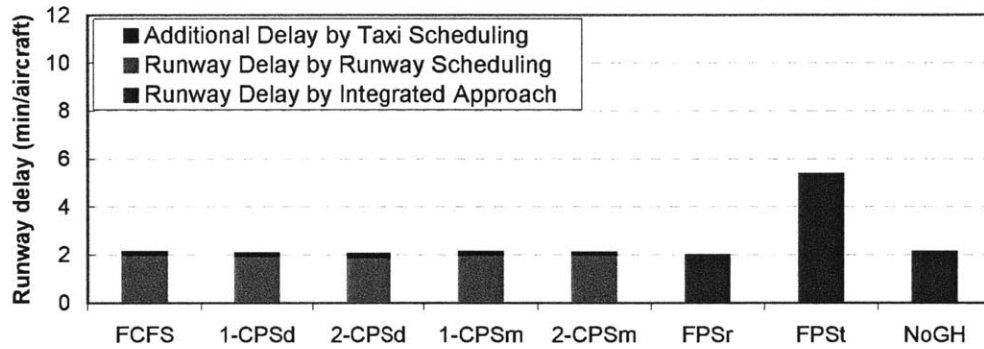


Figure 3-12: Scenario 2: Average takeoff delay per departure

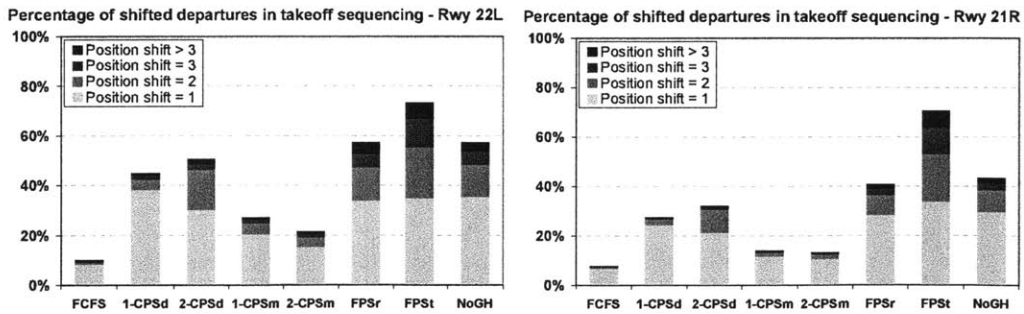


Figure 3-13: Scenario 2: Takeoff order changes from initially estimated sequence

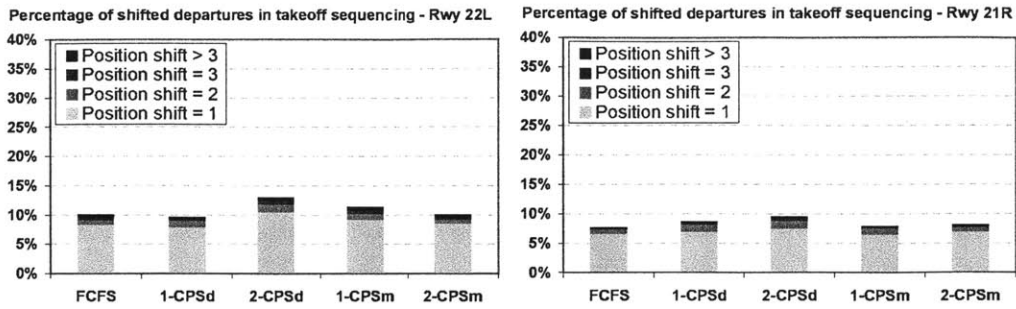


Figure 3-14: Scenario 2: Takeoff order changes between Step 2 and Step 3

3.3.3 Comparison with simulation results

As in Scenario 1, SIMMOD simulations were also run with the optimized pushback times. The simulation results in Figure 3-16 show that the average taxi-out time can be reduced by up to 53 s/aircraft with just pushback time control, compared to the NoGH case. Furthermore, the additional taxi-out time reduction from the RTA control increases from 16 s/aircraft (Figure 3-7) to 30 s/aircraft, because the average runway separation time between takeoffs and the resultant waiting time in the departure queue both increase with more heavy aircraft.

With respect to the new fleet mix ratio, more departures cannot meet the optimal takeoff sequence from the optimization model in the current operational environment. The percentage of departures changing their takeoff sequence between optimization and simulation is also increased, as shown in Figure 3-17.

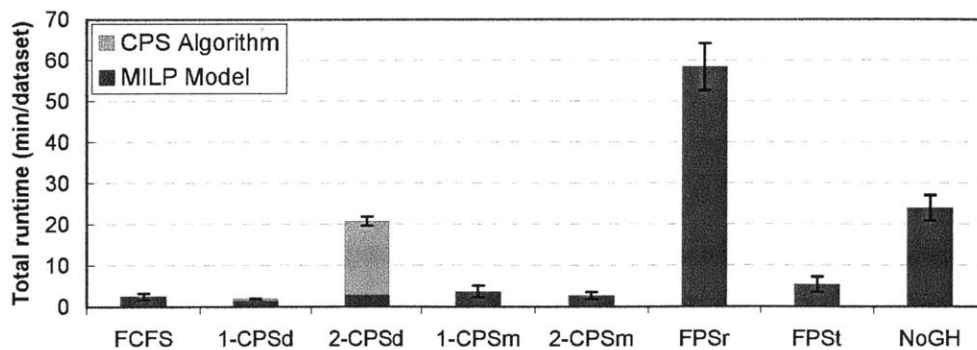


Figure 3-15: Scenario 2: Total runtimes

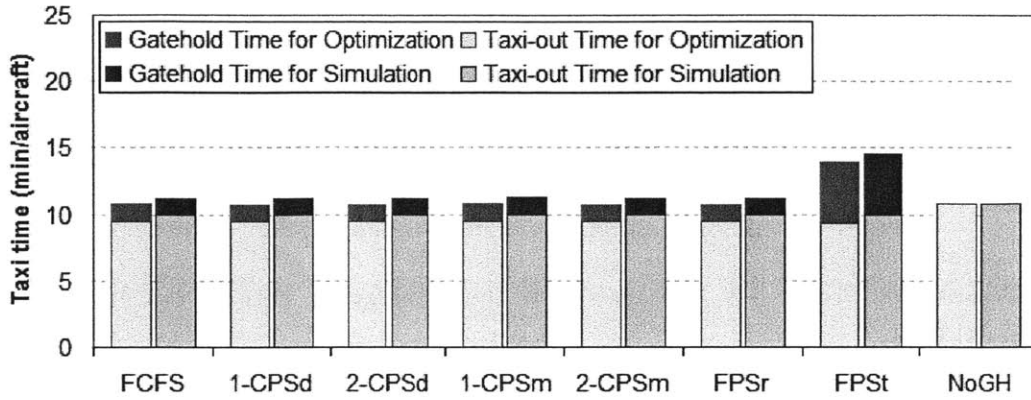


Figure 3-16: Scenario 2: Average taxi-out time comparison between optimization and simulation

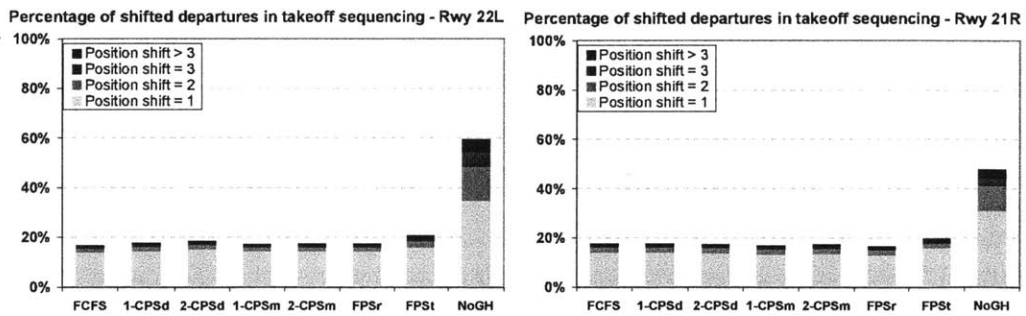


Figure 3-17: Scenario 2: Takeoff order changes between optimization and simulation

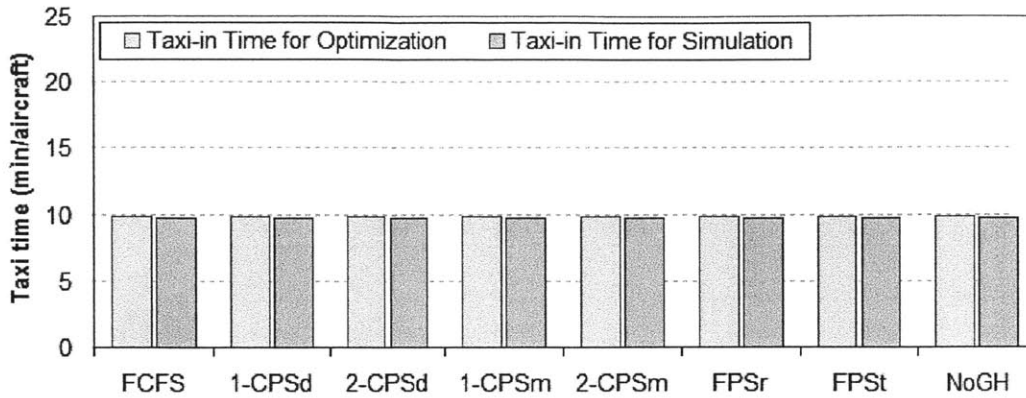


Figure 3-18: Scenario 2: Average taxi-in time comparison between optimization and simulation

Arrivals, however, have no impacts by the fleet mix changes. The simulation results in Figure 3-18 show the almost same taxi-in times as in Figure 3-9. Note that we assume the nominal taxi speed is independent of the weight class of aircraft. Similar to Scenario 1, the average taxi-in times from fast-time simulations are smaller than the optimization results due to the same reasons as explained in the previous section.

3.3.4 Effect of fleet mix ratio

In this section, the effect of fleet mix ratio in the schedule on the airport performance metrics is discussed by comparing Scenario 1 with Scenario 2.

When departures are pushed back as scheduled like the NoGH case, all the metrics indicating the performance of departures are affected. With the increased heavy aircraft portion in the aircraft fleet mix, both taxi-out time and takeoff delay become longer because the average separation time between successive takeoffs increases at the runway, which is recognized as a main bottleneck at an airport. Assuming that the taxi speeds have no difference between aircraft types, arrivals are not impacted by the fleet mix ratio.

When the gate-holding strategy is applied, the average taxi times for departures and arrivals in Scenario 2 are same as the values in Scenario 1. This implicates that

the optimization models guarantee the minimum taxi times for both departures and arrivals, regardless of the fleet mix ratio. However, the average gate-holding time and the takeoff delay are both increased in Scenario 2 due to the existence of more heavy aircraft. The takeoff sequence also changes from the initial estimates more frequently to achieve the better runway schedule.

If the optimized pushback times are used in the current operational environment, the average taxi-out time of departures increases from the optimization results, as shown in the SIMMOD simulation results. This increase becomes larger when the fleet mix has more heavy aircraft because the failure of taxiway conformance to the given RTAs makes the waiting time in the departure queue longer when heavy aircraft show up more frequently.

For the eight different optimization cases, however, it seems that the computational performance is independent of the fleet mix ratio in the flight schedules. All the cases except for the FPSr case showed fast runtimes to find the optimal solutions both in Scenario 1 and Scenario 2.

3.4 Scenario 3: Cyclic high traffic demand with two peak times

3.4.1 More realistic traffic data for Scenario 3

In Scenario 3, we consider more realistic flight schedules with demand fluctuations. According to Figure 3-19, two characteristics can be found in the actual daily traffic demand pattern at DTW. First, the demand to use runways has cyclic periods over time. Second, the departure demand peaks alternate with the arrival's. Also, the hourly demand rate at DTW was around 80 aircraft/h in 2011 when the airport was busy.

From these observations, it is assumed in Scenario 3 that the air traffic demand has two peaks which are 4 hours in length, and varies with time (either 4, 8, 12 or 16 aircraft per 15 minutes for each runway), while the total hourly demand rate is 160

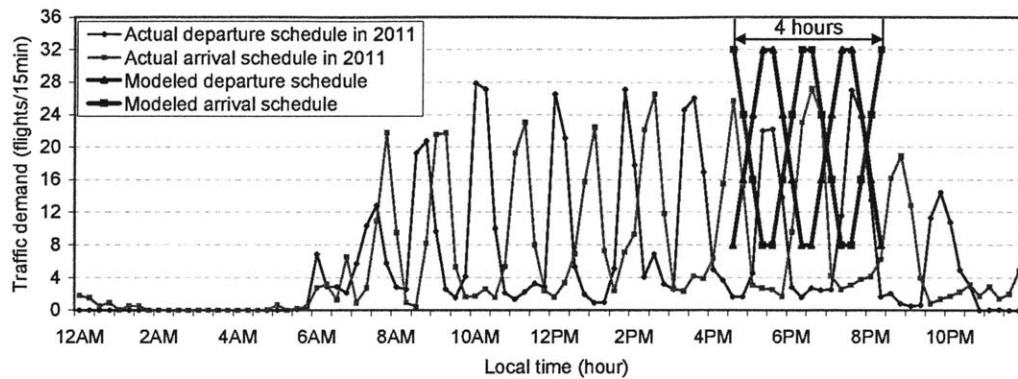


Figure 3-19: Daily traffic demand pattern at DTW in 2011

aircraft/h as before. Arrivals are out of phase with departures. Fleet mix ratio and other assumptions are the same as in Scenario 2 for the purpose of comparison. For this experiment, 27 different sets of flight schedules were generated and optimized in eight optimization cases.

3.4.2 Optimization results

The average values of gate-holding times, taxi-out times, and taxi-in times for the eight optimization cases are summarized in Figures 3-20 and 3-21. Although the time held at the gate significantly increases with the new traffic pattern in Scenario 3, the optimized taxi times are similar to those in previous scenarios. On the other hand, the average taxi-out time in the NoGH case dramatically increases from Scenario 2 because the flights leaving the gates at the scheduled pushback times are stranded in the departure queues during peak times with delay propagation. Therefore, the gate-holding strategy provides relatively more significant benefits when the flight schedule has peak periods like this traffic scenario.

In the five optimization cases from the three-step approach, the takeoff delays in Figure 3-22 are not increased by gate holding, compared to the NoGH case. Even the FPSr case shows the lowest takeoff delay in this scenario. Moreover, the takeoff delay value in this case is lower than the runway delays only by runway scheduling in Step 2 (orange colored bars) for the first five cases in Figure 3-22. This is because

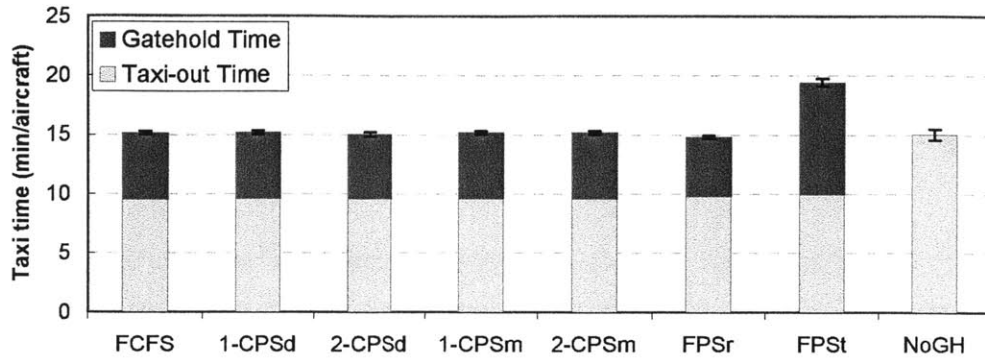


Figure 3-20: Scenario 3: Average taxi-out time per aircraft

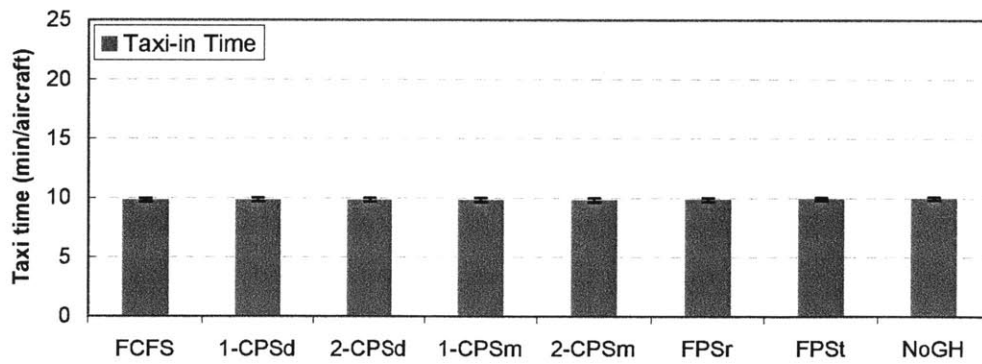


Figure 3-21: Scenario 3: Average taxi-in time per aircraft

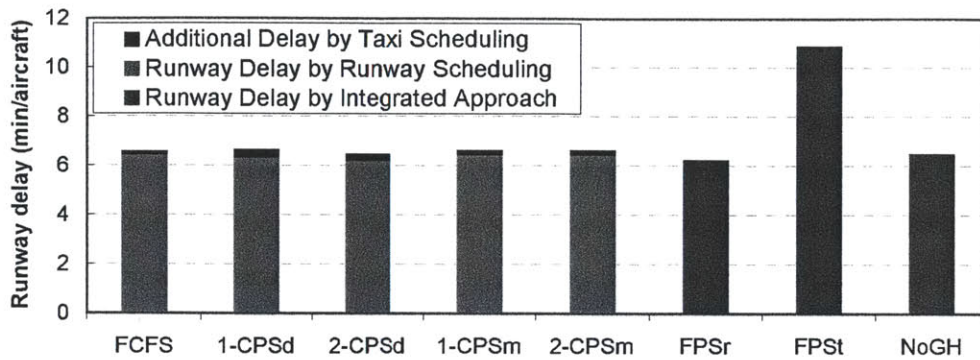


Figure 3-22: Scenario 3: Average takeoff delay per departure

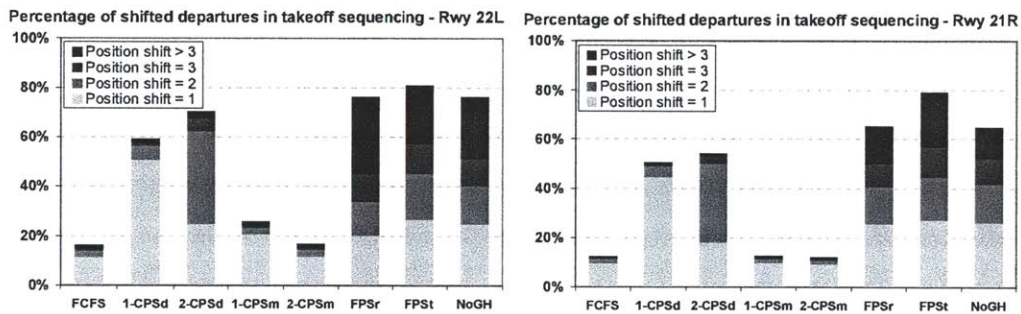


Figure 3-23: Scenario 3: Takeoff order changes from initially estimated sequence

the optimization model in the integrated approach allows unlimited position changes in the takeoff sequence, whereas the runway scheduling algorithm used in Step 2 constrains the maximum position shift up to 1 or 2 slots. It is justified by the graphs in Figure 3-23 showing that the excessive position changes more than 3 position shifts are observed more frequently in the FPSr case. The FPSt case is not compared here because it does not matter about the runway schedule.

The bar graphs in Figure 3-24 show the average values of the largest takeoff delays from the earliest possible takeoff times for two departure runways. The whiskers denote the maximum and minimum values of the largest delays, across the 27 datasets. The takeoff delay is a sum of gate-holding time and ground delay, mainly due to runway separations during peak times. Note that the gate holding is limited to 15 minutes in the optimization for the gate utilization. As seen in the figure, the maximum takeoff delay in every case is higher for Rwy 22L than for Rwy 21R. In the

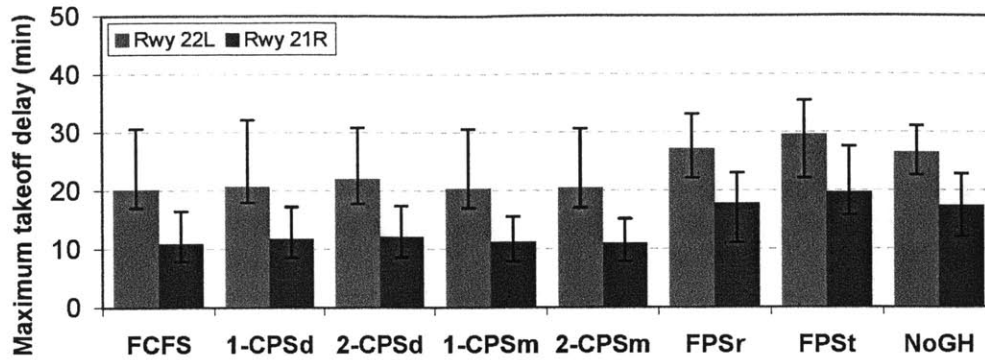


Figure 3-24: Maximum takeoff delay - optimization

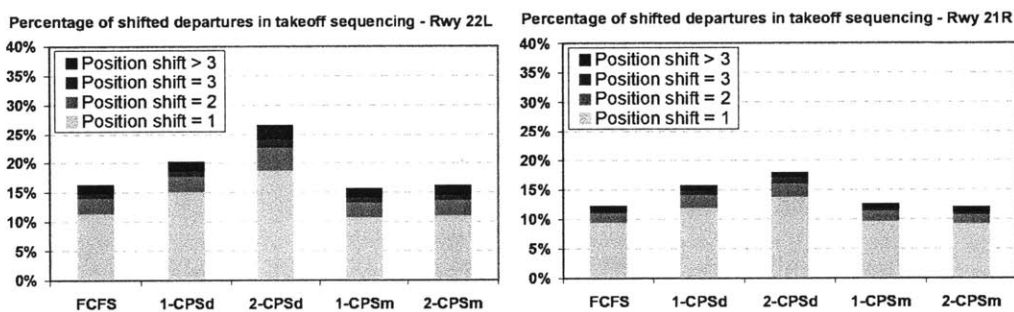


Figure 3-25: Scenario 3: Takeoff order changes between Step 2 and Step 3

three-step approach, the maximum values are always lower than the largest delays in the integrated approach because of the limited position shifts allowed in takeoff sequencing. They are also averagely lower than the NoGH case, even though the gate-holding time is added. If necessary, the maximum takeoff delay can be constrained in the optimization model.

The change of the demand patterns also affects the complexity on the surface. The existence of peak periods in the flight schedule increases a possibility to change the takeoff sequence while taxiway scheduling. In the three-step approach, the percentage of takeoff sequence changes from Step 2 to Step 3 in Figure 3-25 is always higher than the percentage from the previous traffic scenario having no peaks.

Figure 3-26 shows the runway throughput per 15 minutes for both departure runways. The fact that the throughput curves are the same for all the cases, excluding the FPSt case, implicates that the runway throughput is not reduced by optimization,

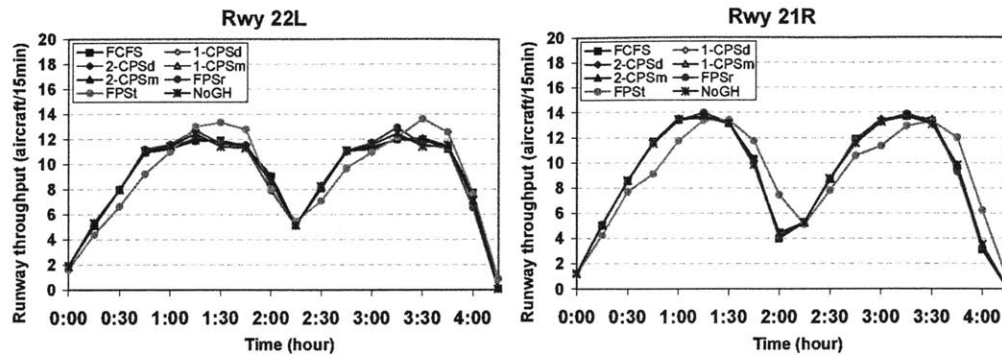


Figure 3-26: Scenario 3: Runway throughput for both departure runways

compared to the NoGH case. However, the FPSt case shows some delays behind the other cases on the throughput curves because of long gate-holding. Note that only Rwy 22L can accommodate heavy aircraft, leading to a little lower throughput on the runway during peak periods.

Figure 3-27 illustrates the number of taxiing aircraft on the ground, N , which is measured every 15 minutes and represents the level of taxiway congestion. In the NoGH case, the number of departures moving out on the ground accumulates along with the growing demand because of the bottleneck on the runway threshold. As a result, long queues are observed for both departure runways during peak times. Due to the presence of heavy aircraft in the fleet mix, Rwy 22L shows relatively higher N values when the traffic is most congested. In this graph, it is found that the taxiway congestion can be mitigated by optimization. All the optimization cases show that N values are maintained less than 10 aircraft for each departure runway. That means the number of departures waiting for takeoff clearances in the queue is minimized. The reason why the Rwy 21R has higher N values in the optimization cases is that its average distance from gates to the runway is longer than that for Rwy 22L.

By integrating the data from Figures 3-26 and 3-27, we can plot the takeoff rates as a function of the number of aircraft taxiing out. In Figure 3-28, the NoGH case follows typical patterns from the actual traffic statistics. On the other hand, we can observe clustering in the other optimization cases. During peak times, the data points are concentrated on the specific region where the maximum runway throughput

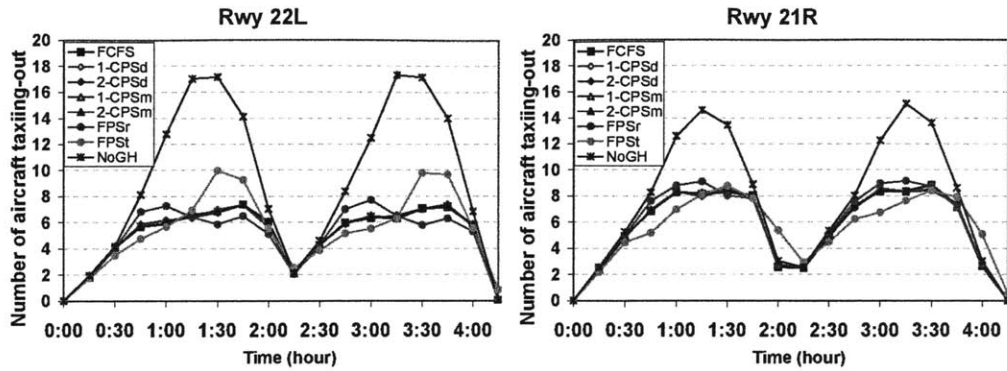


Figure 3-27: Scenario 3: Number of taxiing aircraft for each departure runway

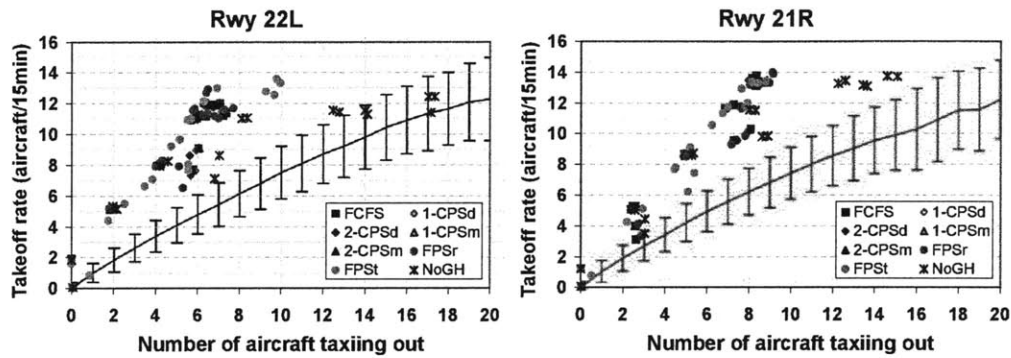


Figure 3-28: Scenario 3: Runway throughput vs. number of taxiing aircraft

intersects with a limited number of taxiing aircraft. In normal traffic, the number of taxiing aircraft is minimized, whereas the takeoff rate is the same as the traffic demand rate.

The plot for the departure throughput as a function of the number of flights taxiing out can be compared with the actual traffic statistics at DTW, as shown in Figure 3-28 [87,113]. The NoGH case basically follows the typical patterns from the actual data, but shows higher takeoff rate at the same surface traffic congestion level in a given runway configuration. The difference between the NoGH case plots and the surveillance data can be explained by the followings: (1) The NoGH case has the optimal takeoff schedules minimizing runway delays rather than heuristics; (2) Tight runway operations to meet the minimum separation between takeoffs, as well as between takeoffs and runway crossings, are assumed; (3) The unimpeded taxi times in

the optimization model ignoring the delay in controller/pilot response are less than the actual ones in the current operations system; (4) The fleet mix ratio in the experiment could be different from the archived operational data; and (5) The actual separation requirement may be different from the rules used in the optimization, especially for Rwy 21R assuming no heavy aircraft allowed. Furthermore, the other optimization cases show much higher runway throughput for the same amount of taxiing aircraft, even for low demand. That can be explained in the same way as described above.

Figure 3-29 shows the total runtimes of the eight different cases for optimizing a 4-hour flight schedule in Scenario 3. Although the FPSr case provides a better optimization result in some metrics than the other cases in the three-step approach, its computational performance is relatively weak. In order to see the effects of the demand pattern, each bar in the graph is subdivided into the average runtime of the CPS algorithm and the average runtimes of the MILP model by optimization time window. Given a time limit of 10 minutes for the optimization solver, the FPSr case often reaches the time limit with a suboptimal solution. In other optimization cases, a significant portion of the total runtime comes from the time windows when the departure demand is peak (4, 5, 6, 12, 13, and 14 in the graph).

3.4.3 Comparison with simulation results

The optimization and simulation results shown in Figure 3-30 compare the average taxi-out times per departure for the eight optimization cases. In Scenario 3, the taxi time savings from optimization are bigger than in the other scenarios. For instance, when we use the optimized pushback time schedule from the FCFS case, we can reduce the taxi-out time by 4.4 min/aircraft and 5.5 min/aircraft by using pushback time control (corresponding to simulation) and RTA control (corresponding to optimization), respectively.

The final takeoff sequences determined by the optimization models can be changed in the simulations, as shown in Figure 3-31. As in the previous scenarios, the NoGH case not using the gate-holding strategy shows the highest percentage of shifted departures in takeoff sequencing for both runways. Compared to the graph in Scenario 2

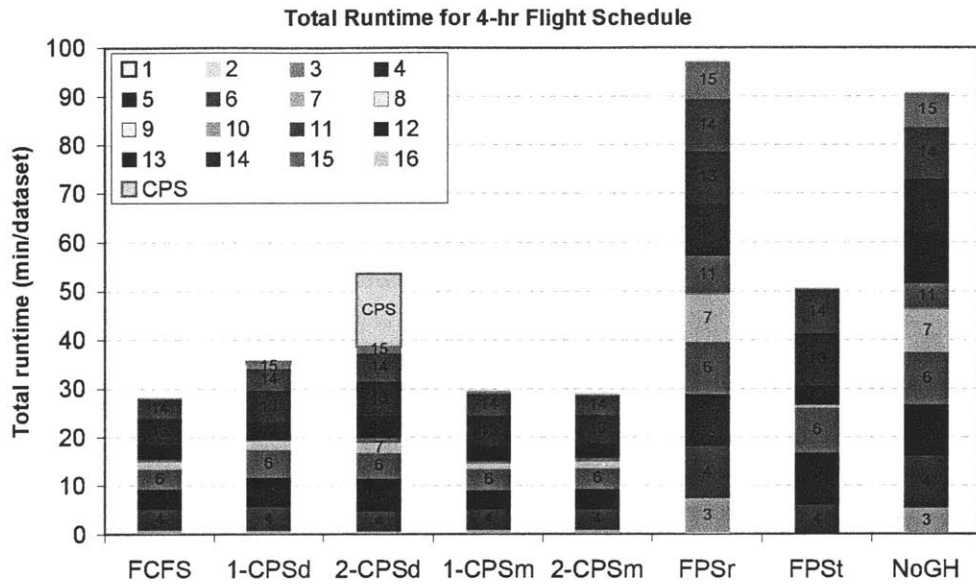


Figure 3-29: Scenario 3: Total runtimes

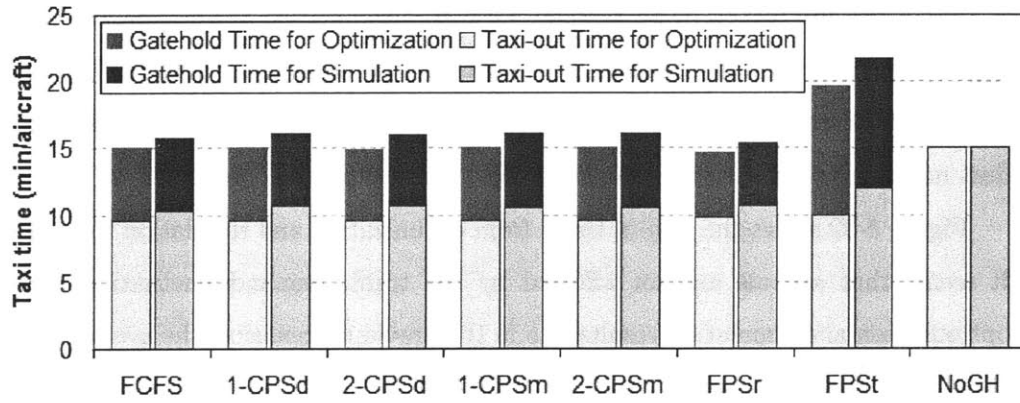


Figure 3-30: Scenario 3: Average taxi-out time comparison between optimization and simulation

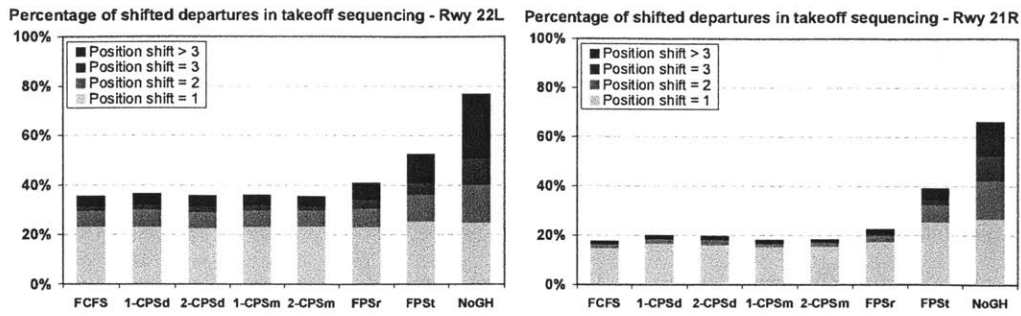


Figure 3-31: Scenario 3: Takeoff order changes between optimization and simulation

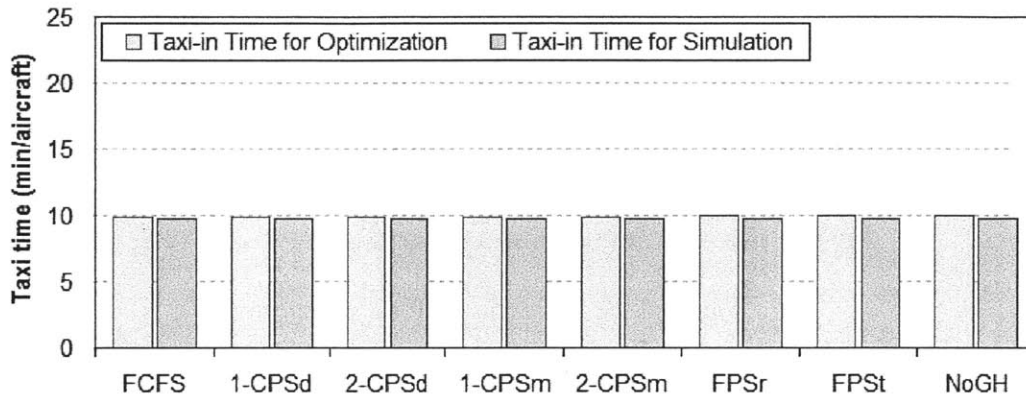


Figure 3-32: Scenario 3: Average taxi-in time comparison between optimization and simulation

(Figure 3-17), the takeoff order changes between optimization and simulation become doubled for Rwy 22L in Scenario 3, whereas the percentage for Rwy 21R does not significantly increase. This indicates again that the existence of heavy aircraft in the fleet mix affects the takeoff sequence conformance of the RTA control.

Figure 3-32 shows the taxi-in times from optimization and simulation for arrivals. It seems that arrivals are not affected by the traffic demand fluctuation both in optimization and simulation results. As in the previous scenarios, the average taxi-in times from fast-time simulations are a little smaller than the optimization results.

The maximum takeoff delays are also compared between optimization and simulation results in Figure 3-33. As seen in Figure 3-24, the maximum delays are always observed at Rwy 22L between two departure runways. The largest takeoff delays are various depending on the flight schedule in a dataset, but their average values are

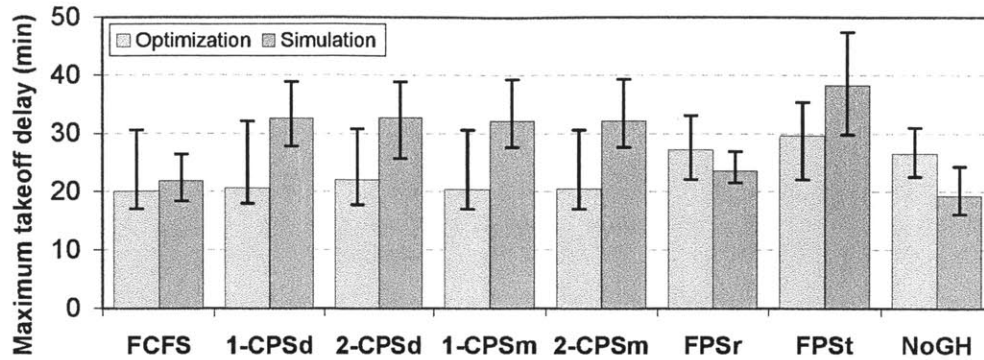


Figure 3-33: Maximum takeoff delay - SIMMOD result

usually higher in the simulation results. In the optimization model, the takeoff times of departures are adjusted by speed control on the ground while taxiing. Without the RTA control, however, some departures can experience much longer delays for takeoffs than expected, by unintentional sequence changes.

Figure 3-34 illustrates the aircraft flows on the airport surface having two different flight schedules, captured from the SIMMOD animations at the same moment. While the upper figure comes from the NoGH case having the initial schedule, the lower one is from the FPSr case with optimized departure schedule. In this figure, we can observe a long queue for takeoffs at Rwy 22L in the NoGH case. On the contrary, the lower picture in Figure 3-34 shows that by controlling the pushback times, we can ensure enough space between flights moving toward the same departure runway, resulting in the significantly reduced waiting times for takeoffs.

3.4.4 Effect of traffic demand fluctuation

This section investigates how the demand patterns in the flight schedule affect the airport performance metrics by comparing Scenario 2 with Scenario 3.

When the gate-holding strategy is not applied to departures, the average taxi-out time of them dramatically increases with the demand fluctuation. During the peak times, the number of flights in the departure queues waiting for takeoffs accumulates, and the waiting time is propagated. By the same token, the takeoff delay also in-

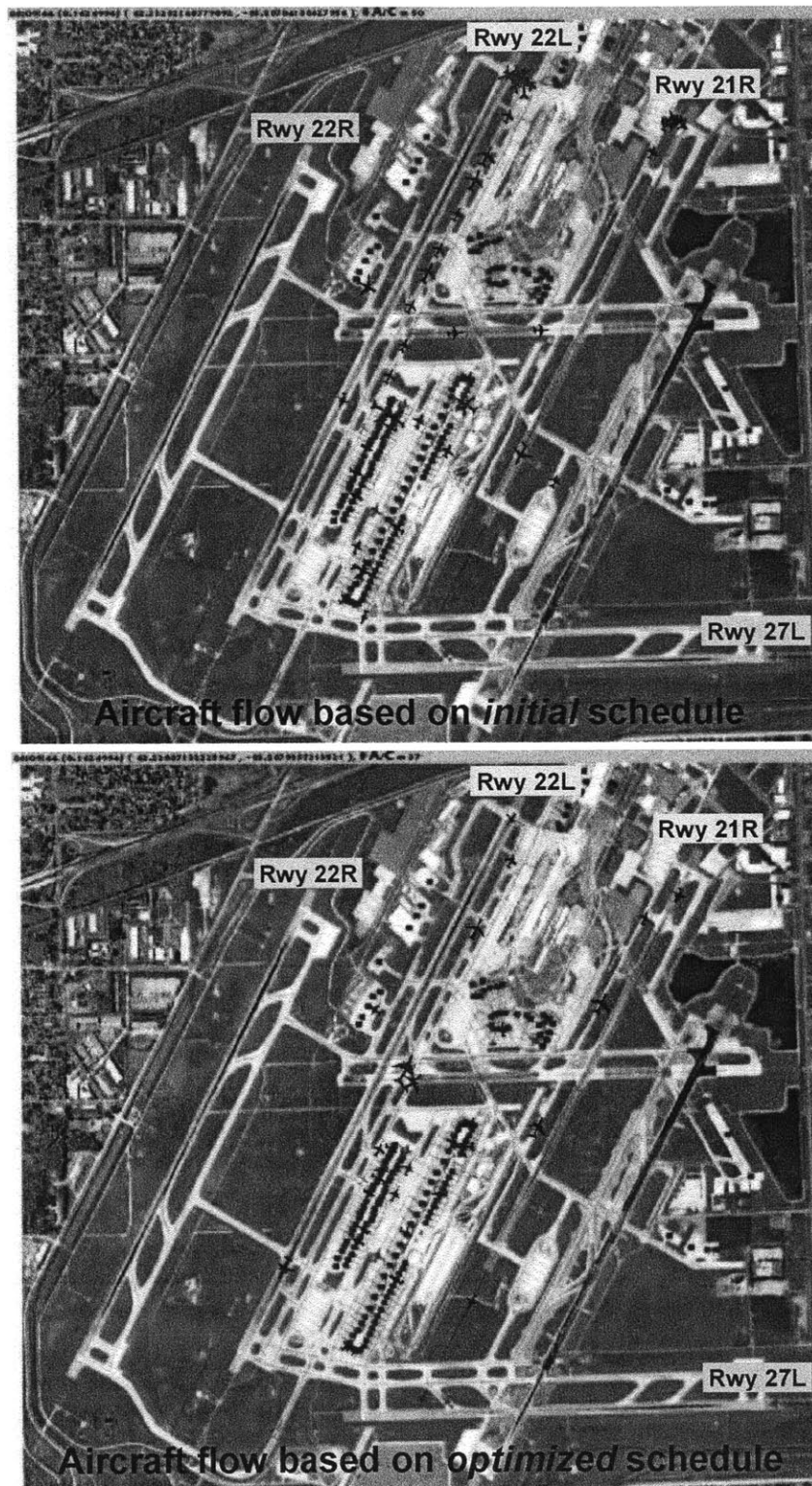


Figure 3-34: Scenario 3: Aircraft flow comparison

creases. This result agrees closely with the on-time performance analyses at New York JFK and EWR airports stating that the more evenly flights are distributed over time, the lower the resulting delays are likely to be [71]. Compared to Scenario 2, the average taxi-in time of arrivals is also increased slightly in Scenario 3 due to the longer holdings for runway crossings.

In Scenario 3, both optimization approaches using the gate-holding policy show the similar taxi-out and taxi-in times to the optimization results for the flattened demand pattern in Scenario 2. However, the demand fluctuation makes both gate-holding time and takeoff delay longer. When the demand for takeoffs exceeds the runway capacity, the takeoff times of departures are delayed exponentially, and its impact is propagated even after peak times. The optimization results in Scenario 3 show the maximum takeoff delay can be longer than 30 minutes.

When the flight schedule has two peaks, the takeoff sequence changes more frequently from the initially estimated order. It seems that the sequence changes occur more frequently at taxiway intersections during the peak times. This is supported by the takeoff order changes from Step 2 to Step 3 in the three-step approach. While optimizing taxiway schedules, the percentage of shifted departures in the sequence is increased by more than 5% in Scenario 3.

According to the simulation results, the taxi-out time difference between the RTA control and the pushback time control becomes bigger in Scenario 3. When the flight schedule includes peak periods, the taxiway is locally more congested, resulting in the longer taxi time, as well as more changes in the takeoff sequence while taxiing without the RTA control. Controlling only the pushback times in the current operations also leads to the increase in the maximum takeoff delay.

When optimizing the realistic flight schedules having peaks and valleys of demand, the total runtimes increase significantly in all the optimization cases. The detailed runtime analyses show that some iterations reach the time limit before obtaining the optimal solution, when the number of departures within 30-minute time window exceeds a threshold. However, the three-step approach is still amenable to real time implementation.

3.5 Conclusions

Through the case study at DTW, two different optimization approaches were evaluated with various airport performance metrics in this chapter. For three high traffic demand scenarios, the optimization results commonly showed significant taxi-out time savings at no expense to other major performance metrics such as takeoff delay, runway throughput and taxi-in time. Compared to the three-step approach, the integrated approach (represented by the FPSr case) had small advantages in terms of taxi-out time and takeoff delay, but its computation time was long.

In the three-step approach, two different objectives were introduced in runway scheduling to study their impacts on the airport performance obtained after taxiway scheduling. Both the minimum makespan solution and the minimum runway delay solution showed the same taxi times, and their runway delay difference in Step 2 was eliminated in Step 3 because of the additional delay induced by taxiway conditions. However, minimizing the makespan would be a better objective when taking controller workload and computational performance into consideration.

A comparison of Scenario 1 and 2 showed that the fleet mix ratio in the flight schedule could affect some performance metrics of departures, including gate-holding time, takeoff delay, and position changes in takeoff sequencing. When the gate-holding strategy was applied, however, the optimized taxi-out and taxi-in times were independent of the aircraft fleet mix.

By comparing Scenario 2 with Scenario 3, we demonstrated that the demand fluctuation had significant impacts on most performance metrics. For the same number of flights, the average taxi times and the takeoff delay dramatically increased due to the existence of peak times when the gate-holding strategy was not used. In other words, the taxiway congestion could be reduced by only flattening the peak demand in the flight schedule. On the other hand, the optimized taxi times from both optimization approaches were not affected by the cyclic traffic pattern, although the takeoff time was more delayed because of longer gate-holding.

We also implemented the fast-time simulations using the optimized pushback times

as inputs in SIMMOD. The comparison between optimization and simulation results showed that the taxi-out time could be significantly reduced by only controlling the pushback times in the current operational environment, although its benefits were smaller than the RTA control.

Chapter 4

Comparison of individual aircraft trajectory-based control and aggregate queue-based control

As mentioned before, there are two kinds of control approaches to achieving efficient surface traffic management: individual aircraft trajectory-based control and aggregate queue-based control. In the previous chapters, we discussed the optimization models to determine the optimal taxiway and runway schedule on the trajectory basis of individual aircraft. In this chapter, we introduce the aggregate queue-based control (N-Control) for departure management. Then, we describe two trajectory-based control approaches (RTA control and PbT control), depending on the technology readiness level in the operational environment.

These control approaches are implemented with the same traffic scenarios for comparison and evaluated in terms of various airport performance metrics largely categorized into gate, taxiway, and runway.

The gate-holding strategy commonly used in these departure control approaches can cause the gate conflict between a gate-held departure and an arriving aircraft assigned to the same gate. The frequency of gate conflicts is analyzed for investigating their impacts on the surface traffic. Then, several possible solutions to avoid the gate conflicts are proposed, and the effects of the solutions are evaluated.

4.1 Aggregate queue-based control for departure management (N-Control)

For efficient surface traffic management, some control approaches deal with departing aircraft in an aggregate manner rather than using the trajectory of individual aircraft. From this point of view, the traffic flow of aircraft on the surface is taken into account more importantly than the detailed movement of each flight. Also, aircraft counts in a specific control volume at the airport are significantly used for the optimal control. The models using this aggregate control approach are called *Eulerian* models. The objective of the Eulerian models is to send the flights released from gates to runways efficiently without starving the runways. In these models, taxiway area and runway departure queues are considered as specific control volumes of a queueing system, and the runway is recognized as a bottleneck of the system. In order to reduce congestion on the ground, we can limit the number of taxi-out aircraft entering the queueing system below a specified control parameter by controlling the pushback times of departures. This approach is called the *N-Control* strategy.

N-Control is an initiative to reduce taxi delays and emissions in the departure process while maintaining high departure throughput, which is motivated by the fact that the takeoff rate is saturated when the number of taxi-out aircraft, N , is greater than a threshold, N^* [111]. N-Control was initially proposed in the Departure Planner project [49], and its variants have been discussed in later studies [30,31,103]. The main idea of N-Control was derived from an observation of the airport performance curve showing the departure takeoff rate as a function of the number of aircraft taxiing out. Figure 4-1 illustrates a typical form of the airport performance curve. The curve is drawn based on the statistical data at an airport. The pattern of the curve can be different depending on the airport operational conditions such as runway configuration, weather, and traffic characteristics. In Figure 4-1, the takeoff rate increases when more departures are pushed back from their gates onto the surface because the demand for takeoff grows. However, as surface traffic increases further, the runway system reaches its capacity limit and the takeoff rate is saturated eventually. That is,

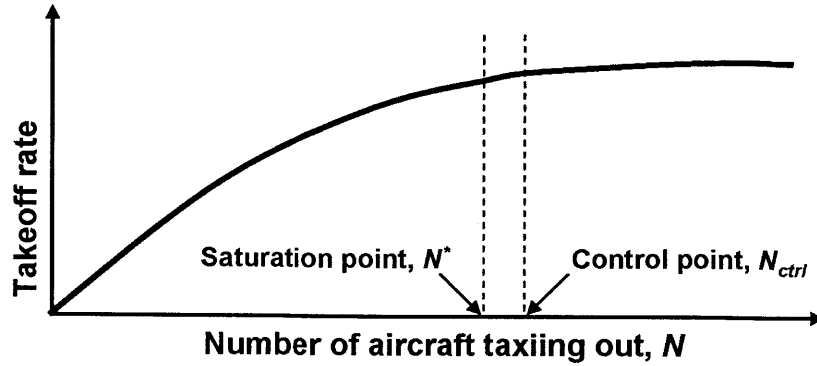


Figure 4-1: Typical airport performance curve for takeoff rate as a function of the number of aircraft taxiing out

if the number of taxi-out aircraft exceeds a saturation point, additional pushbacks of departures will not help the runway throughput increase, but increase only taxiway congestion and taxi-out delays [114].

From this observation, we can find that it would be better to hold some departures at gates instead of pushing them back when the surface congestion level is higher than a control point, denoted by N_{ctrl} . Accounting for possible variations from the averaged curve, the control point is generally chosen slightly higher than the expected saturation point (N^*) to maintain high runway throughput, but not too high to increase the surface congestion level.

Then, the N-Control method can be applied to the airport operations heuristically. To figure out the congestion level on the surface, the number of active departures on the taxiway, N , is counted. If the total number of aircraft on the ground exceeds the chosen control point, the gate-holding strategy is applied to the departures ready to push back. That is, controllers do not allow any more aircraft requesting pushback clearance to move out onto the taxiway until the number of active aircraft on the surface drops below the threshold. In fact, a similar policy has already been used by air traffic controllers heuristically during excessively congested situations.

By introducing the N-Control policy, we can expect to reduce the taxi times, including the waiting times in departure queues, and the corresponding fuel burn and emissions on the surface, while maintaining the runway utilization [112]. Since

the N-Control approach activates the gate-holding strategy only when the surface traffic is congested, it requires minimal modifications to the current procedures. Furthermore, the additional workload for controllers to implement this approach is also limited [116].

In the N-Control approach, however, interaction with arrivals or other departures on the taxiway is not considered. The possible holds at intersections or runway crossings to avoid conflicts can lead to adverse effects on other performance metrics such as increased taxi-in time and reduced runway throughput. In addition, actual takeoff time of a gate-held flight can be delayed from the initially estimated takeoff time due to the same reason. Unless the maximum gate-holding time is restricted, several departing aircraft may experience too long waiting time at gates, sometimes resulting in conflicts with arriving aircraft assigned to the same gate. The departure rate predictions along with the taxiway congestion are based on empirical data, which is specific to the runway configuration at a given airport. If a new runway configuration is introduced or the airport layout changes due to some reasons like runway closures and building a new terminal, therefore, the choice of queue control parameters would be difficult until statistical data are accumulated.

Various surface congestion management methods similar to the N-Control concept have recently been applied to several busy airports in field tests or automated simulation environments. A specific Surface Congestion Management (SCM) approach at New York JFK airport was developed by PASSUR Aerospace, Inc. and assessed in terms of taxi-out time reductions, fuel savings, and emissions reduction [95]. The Collaborative Departure Queue Management (CDQM) concept was also tested at Memphis airport (MEM) [27]. Through the active participation of air traffic controllers, the human-in-the-loop (HITL) simulations of the Spot And Runway Departure Advisor (SARDA) concept were performed for Dallas/Fort Worth airport (DFW) [64,65,75]. Another heuristic modification of N-Control, called the Pushback Rate Control, was developed for easier implementation and successfully tested at Boston Logan International Airport (BOS) [116]. In Europe, the Departure Management System (DMAN) concept [25] was tested at Athens international airport (ATH) [110]. The detailed

implementation techniques are dependent on the target airport, but the fundamental concept of the departure control approaches is same, which is to control the departure queue size by holding a limited number of departure flights in a holding area during the congested conditions.

4.2 Individual aircraft trajectory-based control

4.2.1 RTA control

While the aggregate queue-based control model manages the pushback times of excess departures only, the individual aircraft trajectory-based optimization model finds the optimal taxiway schedules along the given taxi routes, as well as the controlled pushback times of all departures and the optimal takeoff sequences at runways. This approach determines the Required Times of Arrival (RTAs) of taxiing aircraft at significant control points on the taxiway. These control points include gates or spots to start taxiing, intersection nodes between two taxiways, holding areas for runway crossings, and runway thresholds for takeoffs.

By this *RTA control*, we can obtain the minimum taxi times that can be obtained in a given flight schedule, while maintaining the maximum runway throughput. In addition, we can minimize the frequency of possible conflicts at taxiway intersections and reduce operational uncertainties. To maximize the benefits from the RTA control, the gate-holding strategy is aggressively applied to most departing flights, even in low traffic conditions. In this approach, the arbiter of aircraft movements on the surface is centralized from the cockpit of individual aircraft to the control tower, which can lead to the better airport management for the system objective.

To meet the RTAs on the surface, the taxi speed of aircraft on the ground is controlled. However, it is hard to control the taxi speed of individual aircraft as planned. To realize the RTA control, therefore, we need enhanced Communications, Navigation, and Surveillance (CNS) technologies for the communication between controllers and cockpit crew and the taxi conformance monitoring [35]. The relevant advanced

equipment should be equipped both in aircraft and at airport traffic control towers. New operational procedures for Surface Trajectory-Based Operations (STBO) are also required, for which additional controller workload and training are expected. Moreover, since the existing trajectory-based optimization usually needs long runtime to find an optimal solution, it also requires a better computational performance for the faster response speed to the schedule updates.

4.2.2 PbT control

In the current operational environment, it is not possible to implement the RTA control yet. Controllers usually control the departures ready to move out from their gates by giving them pushback clearance. So, even though the RTAs from the trajectory-based optimization model ensures the minimum taxi time and takeoff delay, the only control points where we can control departures in the current system would be gates. From a practical perspective, therefore, we can consider using the optimal pushback times (PbT) only from the optimization model, instead of meeting all the RTAs along the taxi routes. This interim control approach will be called *PbT control* in this thesis because in this approach, controllers can control the pushback times only at gates among the RTAs at all control points on the surface.

Compared to the RTA control, it is expected that the benefits from the departure management will be reduced since the interaction with other flights is not properly handled on the taxiway by controllers as optimized. During peak periods, the takeoff sequence may change from the optimal sequence while taxiing, leading to the reduced runway throughput. On the other hand, the PbT control would still result in shorter taxi time than the N-Control because more departures are held at gates, instead of waiting in departure runway queues, to have the separation required for successive takeoffs. The more aggressive gate-holding in the PbT control, however, may increase the workload of controllers.

In the previous research at DTW, the PbT control has been compared with the N-Control by using the pushback time schedule optimized by the trajectory-based optimization model as the input to the queuing model [87]. The result showed that

the runways were sometimes under-utilized ending up in additional takeoff delays because some departures arrived at the runways at non-scheduled times. However, the experiment was for the 1-day flight schedule at the current traffic demand level. In the heavy traffic scenarios overloading runways frequently, the PbT control is expected to get more benefits on taxi times than the N-Control approach having a reduced control parameter, while maintaining similar runway throughput. As in the N-Control approach, however, the PbT control also has some fundamental limitations in terms of control input because its control points are not the departure runway queues, but the gates, whereas the main bottleneck of the surface traffic is the runway.

4.3 Implementation of different departure control approaches

The different control approaches to managing the surface traffic congestion described above are implemented and evaluated with the same traffic scenarios in this chapter. Since these control approaches are based on the different framework, the details for their implementation, which will be described in this section, are not same. To make the comparison reasonable, they have the identical assumptions on the fundamental rules in operations, such as separation requirements, nominal taxi speeds, and schedule update cycle.

4.3.1 Implementation of N-Control

Aggregate queue-based control of departures is a simple strategy that can easily be implemented under current operational procedures [114]. In this chapter, the N-Control policy is performed as the following procedure. Before starting each 15-minute period,

- 1) We count the number of departing flights currently active on the ground. This number can be obtained by calculating the difference between the total number of departures already pushed back and the total number of departures that took off

by that moment. In actual operations, the number can be counted through visual observations.

2) The takeoff times of departures within 15 minutes from the starting moment are estimated from the scheduled pushback times, their unimpeded taxi-out times along the given taxi routes, and the imposed separation requirements for consecutive take-offs. From the predicted takeoff times, we can also calculate the expected departure throughput in the next 15-minute time window.

3) The difference between the number of active departures on the surface and the expected departure throughput is calculated. If the number of active departures is less than the throughput, the gate-holding will not be applied to any flights ready to push back. If not, this difference represents the number of active flights that are expected to remain on the ground through the next 15-minute period. Using the estimated takeoff times, we can pick out which flights will remain on the ground.

4) The difference between the number of these flights and the given queue control parameter N_{ctrl} provides us with the additional number of flights allowed to push back as scheduled in the next 15 minutes.

5) The excess flights in the next time window should be rescheduled with the introduction of gate holding. The pushback times of them will be shifted to the next takeoff times in the initial pushback time order. In this way, the number of active departures on the surface can be maintained below the given N_{ctrl} value.

In case that there are multiple departure runways at the airport like DTW, DFW, and ATL, this control approach can be applied to each departure runway by separating the whole departing flights into several flight groups assigned to the same runway [87].

The final pushback times adjusted by N-Control will be used as input data in the fast-time simulations using SIMMOD for evaluation. The choice of the queue control parameters is critical to the simulation results. If they are too high, they will have no impacts on the airport performance. In contrast, if they are too low, they will introduce significant delays [114]. To prove this anticipation and determine the optimal parameter in the given traffic conditions, various queue control parameters will be tested in the simulations.

4.3.2 Implementation of RTA control and PbT control

The input data used in the RTA control can be obtained by solving the optimization model for taxiway and runway scheduling described in Chapter 2. The individual aircraft trajectory-based optimization provides the Required Times of Arrival (RTAs) at all control points on the surface, including pushback times at gates, takeoff times of departures, gate-in times of arrivals, and passage times at intersections. However, following the given RTAs precisely is the ideal case because the optimization is based on the deterministic model. Due to various uncertainties inherent in surface operations, it is almost impossible to meet the RTAs from the optimization model exactly under the current operational environment. In the evaluation of the departure control approaches, the RTA control will show the best solution which can be achieved with the given flight schedule by departure control, compared to the other approaches.

When implementing the RTA control, the optimization model determines the RTAs of departures and arrivals within the 30-minute time window. Also, the optimal flight schedules are updated every 15 minutes, which is same as the N-Control's update cycle. Once the flights are already optimized in the previous iteration, the RTAs of them are assumed to be fixed. In this way, we can prevent some departures from experiencing significant delays by repeated rescheduling and reduce the workload of controllers and pilots required to meet the capricious RTAs. This assumption will also help maintain the trajectory conformance at an acceptable level.

The PbT control is the realizable version of the RTA control in the current surface operation system. These two control approaches share the same pushback times of departures, as well as the same landing times of arrivals. The PbT control can be simulated in SIMMOD by inputting these times that flights begin going into the taxiway system. In the SIMMOD simulations, the passage times at intersections or runway crossings are not controlled. This is similar to the current operational conditions.

The outcome from the implementation of a control approach can be different from the expectations because of the complexity of the taxi-out process. There may

exist tradeoffs between two outputs indicating the airport performance. Therefore, the actual impacts of the departure control strategy should be assessed in terms of various airport performance metrics such as taxi time, takeoff delay, and runway throughput. This assessment will be described in the following section.

4.4 Airport performance comparison

In this section, we compare the different control policies for surface congestion management described in the previous section. To see the benefits from the gate-holding strategy, another case using the scheduled pushback times is also simulated. This ‘do-nothing’ approach is called *No control* in this thesis. So, there are four different control approaches, namely, RTA control, PbT control, N-Control, and No control. The N-Control approach has several derivatives depending on the queue capacity control parameters assigned to the departure runway queues.

These control approaches should be compared with each other using the same flight schedule. In this section, they are applied to the 4-hr flight schedules at DTW as for Scenario 3 used in the previous chapter. In this evaluation, the departure queue capacity control parameter N_{ctrl} used in the N-Control approach ranges from 7 to 15 aircraft per queue. Although the two departure runways at DTW (Rwy 22L and 21R) may have different saturation points N^* , it is assumed that they have the same control parameter for each case in order to see the tendency of the performance metrics with respect to the parameter. Note that when the N_{ctrl} value is too low, the gate-holding time applied to a departure becomes so long that the gate conflict with an arrival using the same gate can happen.

The simulation results from these control approaches are then evaluated in terms of various airport performance metrics. These performance metrics can be categorized into three major airport elements: gate, taxiway, and runway. At gates, the number of gate-held flights, the average gate-holding time, and the maximum holding time are compared. For the taxiway area, the average taxi-out time of departures, the average taxi-in time of arrivals, the number of taxiing aircraft, and the number of stops while

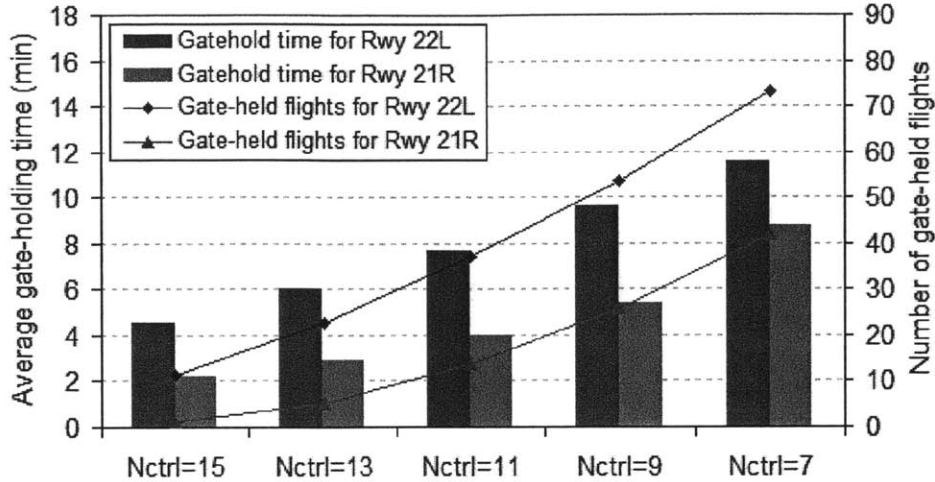


Figure 4-2: Average gate-holding time and number of gate-held flights

taxiing are evaluated. Lastly, the runway performance such as runway throughput, the average takeoff delay, and the maximum takeoff delay is also investigated.

4.4.1 Gates

First, we focus on the relationship between the gate holding level and the queue control parameter in the N-Control approach. Figure 4-2 shows the number of gate-held flights and the average gate-holding time of them by departure runway, depending on the N_{ctrl} values from 15 to 7. As the N_{ctrl} value becomes lower, more flights are held at gates for a longer time. For the same N_{ctrl} value, flights toward runway 22L are affected more heavily by the gate-holding strategy because a longer waiting time is expected in the departure queue for the runway. Note that Rwy 22L only can allow the heavy type of aircraft to take off at DTW due to the runway requirement about the minimum takeoff roll distance. When $N_{ctrl} = 7$, total 36% departures are held at gates for 10.5 minutes on average. As a reference, the FCFS case, a typical trajectory-based optimization case, makes 88% flights to be held at gates for 6.3 minutes.

Table 4.1 provides us with more detailed gate-holding data depending on the N_{ctrl} value for both departure runways. Both the total gate-holding time and the number

N_{ctrl} value	15	13	11	9	7	FCFS
Total gate-holding time (min)	53.9	151.2	341.2	654.0	1211.4	1766.3
Number of gate-held flights (out of 320 departures)	12.5	27.7	50.9	79.3	115.0	280.6
Avg. gate-holding time (min/aircraft)	0.17	0.47	1.07	2.04	3.79	5.52
Avg. gate-holding time of flights held (min/aircraft)	4.3	5.5	6.7	8.3	10.5	6.3
Max. gate-holding time applied (min)	9.5	11.9	15.0	28.1	33.2	15.0

Table 4.1: N-Control approach gate-holding result depending on N_{ctrl}

N_{ctrl} value for Rwy 22L	15	13	11	9	7	FCFS
Total gate-holding time (min)	51.7	136.7	286.6	514.3	844.1	1229.4
Number of gate-held flights (out of 160 departures)	11.5	22.8	37.2	53.5	73.1	145.0
Avg. gate-holding time (min/aircraft)	0.32	0.85	1.79	3.21	5.28	7.68
Avg. gate-holding time of flights held (min/aircraft)	4.5	6.0	7.7	9.6	11.6	8.48
Max. gate-holding time applied (min)	9.5	11.9	15.0	28.1	33.2	15.0

Table 4.2: N-Control approach gate-holding result for Runway 22L

of gate-held flights out of 320 departures in the 4-hr flight schedule dramatically increase as the control parameter decreases. However, the effects of the N-Control approach at gates are much less than the trajectory-based optimization approach. The FCFS case in the last column of the table shows 1.5 times longer gate-holding time and 2.4 times more gate-held flights, compared to the most aggressive case in the N-Control approach ($N_{ctrl}=7$). However, its average gate-holding time per gate-held flight is about 60%. This suggests that N-Control holds the excess flights only at gates, whereas the individual aircraft trajectory-based control approach uses the gate-holding strategy more extensively. As a result, when the control parameter is low, the maximum gate-holding time of the excess flights can be too long unless there is a proper constraint to avoid gate conflicts.

In Tables 4.2 and 4.3, the gate-holding results by N-Control are divided into each departure runway. When the same N_{ctrl} value is applied, the departures assigned to the busier departure runway, Rwy 22L, are more frequently held at gates, and its gate-holding time is longer. In the gate-holding results from the trajectory-based control approach, a similar tendency is seen.

N_{ctrl} value for Rwy 21R	15	13	11	9	7	FCFS
Total gate-holding time (min)	2.2	14.5	54.6	139.7	367.3	536.9
Number of gate-held flights (out of 160 departures)	1.0	5.0	13.6	25.8	41.9	135.6
Avg. gate-holding time (min/aircraft)	0.01	0.09	0.34	0.87	2.30	3.36
Avg. gate-holding time of flights held (min/aircraft)	2.2	2.9	4.0	5.4	8.8	4.0
Max. gate-holding time applied (min)	6.9	8.9	10.9	12.9	14.9	15.0

Table 4.3: N-Control approach gate-holding result for Runway 21R

4.4.2 Taxiways

The average taxi-out times from four different departure control approaches are illustrated and compared each other in Figure 4-3. In this evaluation, the FCFS case is used as a typical trajectory-based control. As described in the previous section, the optimization result of the FCFS case and its simulation result from SIMMOD runs correspond to RTA control and PbT control, respectively. The N-Control method is subdivided with various queue control parameters N_{ctrl} between 7 and 15 in the figure. According to the graph, the taxi-out time savings from the trajectory-based control approaches (RTA control and PbT control) are much higher than the queue-based control, even though a very low queue capacity control parameter is applied. These additional savings are obtained by controlling more departures aggressively. As seen in Table 4.1, the trajectory-based control holds most flights at their gates even when the taxiway is not congested, aiming to prevent almost all flights from waiting in departure queues. On the other hand, the N-Control method allows some departures to wait in the queues and becomes activated only when the runway is saturated.

In the N-Control cases, the average taxi-out time decreases as the N_{ctrl} value decreases. When the N_{ctrl} value is greater than 7, the taxi time savings translate to the holding times at gates (see the dark color bars in Figure 4-3), resulting in the same takeoff time as the ‘No control’ case (see a dotted line matched with the top of the last bar in Figure 4-3). When $N_{ctrl}=7$, the taxi time is reduced by 2.9 min per aircraft, but its takeoff is delayed by 0.8 min due to excessive gate-holding. Compared to the PbT control, the N-Control case with $N_{ctrl}=7$ shows similar takeoff delay, but

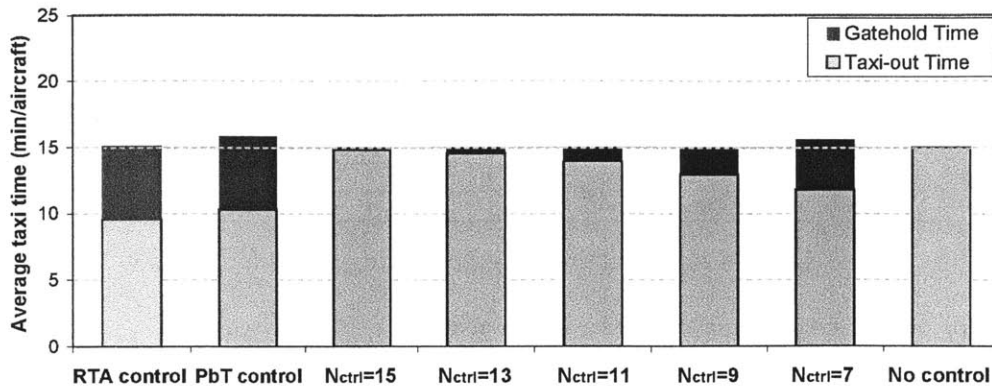


Figure 4-3: Taxi-out time comparison

its taxi-out time is longer. This observation tells us that setting the queue capacity on N-Control too low can compel departures to stay at their gates for a long time, resulting in takeoff delay. In addition, the usage of the low queue control parameter can increase the possibility of gate conflicts, which will be discussed later.

Since there are two departure runways at DTW, the taxi-out time comparison in Figure 4-3 can be broken down by runway, as shown in Figure 4-4. Although the number of flights using each runway is same, the absolute values of taxi-out times and gate-holding times are not the same because of their different traffic characteristics. For example, the average unimpeded taxi-out times from gate to runway are different as Rwy 22L is closer to the main terminal. Also, there is a distinct difference between their fleet mix ratios since Rwy 21R cannot accommodate heavy aircraft due to its shorter length. However, the trend among control approaches is same for both runways. That is, the RTA control obtains the most taxi time savings without takeoff delays; the PbT control shows a significant taxi time reduction with small takeoff delays; and the taxi-out time decreases as the N_{ctrl} value on the N-Control decreases, but the amount of taxi time reduction is smaller than the trajectory-based control approaches. By comparing the two bar graphs in Figure 4-4, we can also observe that flight schedules having more heterogeneous fleet mix get more benefits from departure control.

Taxi-in times of arrivals are not affected by the departure control strategy, as

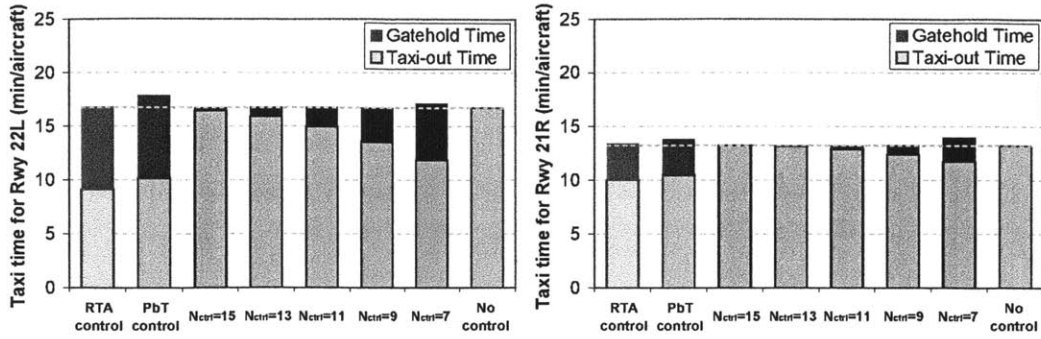


Figure 4-4: Taxi-out time comparison for each departure runway

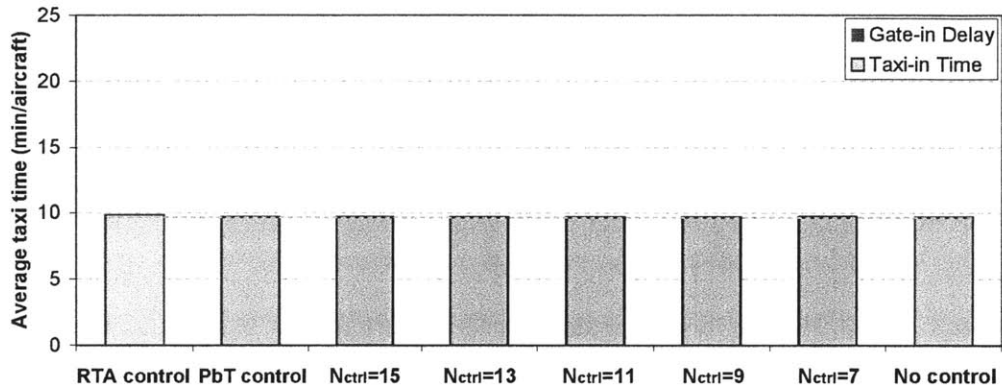


Figure 4-5: Taxi-in time comparison

long as a gate conflict does not happen. Figure 4-5 shows taxi-in times from various departure control approaches, including N-Control cases with various queue control parameters. While the average taxi-in time increases a little in the trajectory-based control cases, it is not affected by the aggregate queue-based control. When the N_{ctrl} value is very low, however, some departures are excessively held at gates, leading to gate conflicts and the corresponding gate-in delays for arrivals assigned to the same gates.

As an indicator of the taxiway congestion, the number of aircraft taxiing out to each departure runway, N , is shown in Figure 4-6, depending on the threshold capacity, N_{ctrl} . In the graphs, the N values of the N-Control cases are lowered from the NoGH case during peak times and maintained less than the corresponding queue capacity control parameters. This means the taxiway congestion can be mitigated

to an acceptable level by N-Control. The curves for two departure runways have different forms in terms of the maximum number of taxiing aircraft and the duration of peak times. This difference stems from their different fleet mix ratios, making Rwy 22L more congested. Trajectory-based control cases (RTA control and PbT control) from the FCFS optimization case are plotted as well in Figure 4-6 for comparison. They show the similar level of N values to the N-Control with $N_{ctrl}=7$ for Rwy 22L and with $N_{ctrl}=9$ for Rwy 21R.

Figure 4-7 compares the total numbers of stops while taxiing on the ground when a 4-hr flight schedule is implemented with different departure planning. The number of stops is obtained by counting the number of times that the aircraft is held on the taxiway for a while in the SIMMOD simulation. Depending on the locations where aircraft stop, these stops are categorized as one of the followings: holding in departure queues, holding on taxiways, and holding for runway crossings. When the departure schedule is not controlled as in the NoGH case, most stops occur in departure runway queue areas. The number of holdings in departure queues is sensitive to the queue control parameter on N-Control. As the N_{ctrl} value decreases from 15 to 7, the number of holdings in the queues also decreases. Secondly, holdings due to taxiway separation are induced by various reasons related to the operational safety on the ground movement, including the minimum distance between two aircraft entering the same taxiway link, the minimum ground separation constraints, and vehicles traveling in the opposite direction on a no-passing link. Regardless of the queue control capacity

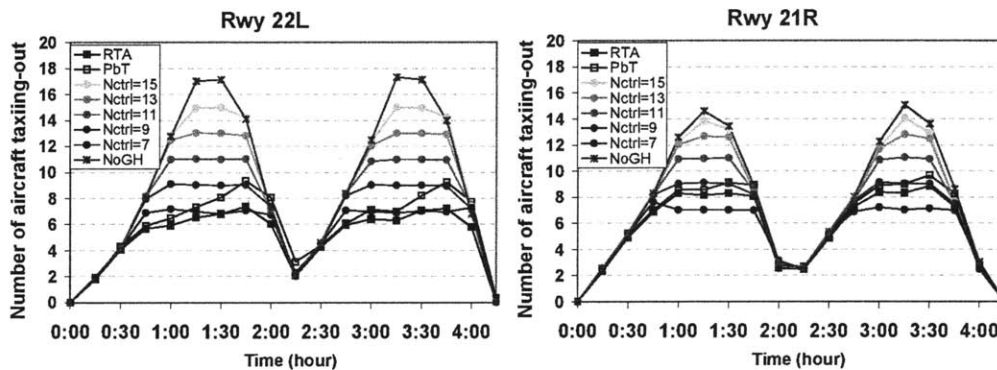


Figure 4-6: Number of taxiing aircraft for each departure runway

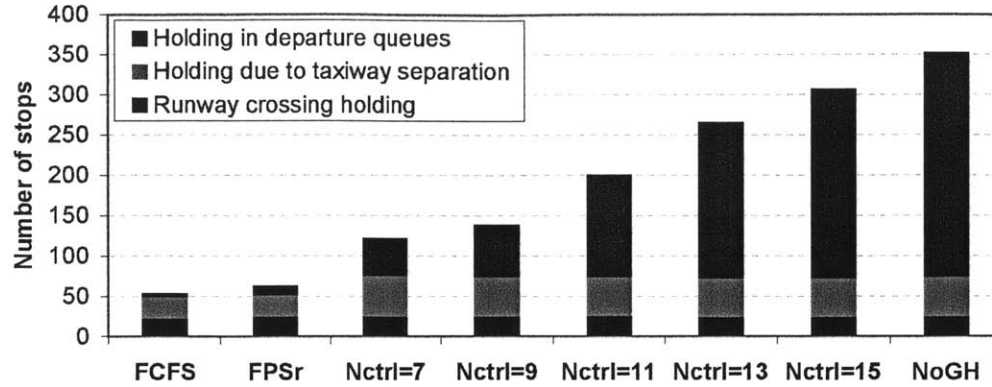


Figure 4-7: Number of stops during taxiing

on N-Control, the total number of holdings due to taxiway separation is constant, whereas the optimization cases (FCFS and FPSr) for the PbT control show fewer stops. Lastly, runway crossing holdings are usually applied to arrivals which need to cross the active departure runways, so it seems that the number of holdings is independent of the departure control method.

4.4.3 Runways

To evaluate the runway performance, runway throughput curves for departure runways are illustrated in Figure 4-8. Compared to the NoGH case, there are no significant impacts on runway throughput by any departure control strategies during non-peak times. However, some peak points in the runway throughput curves are flattened by N-Control. Especially when $N_{ctrl} = 7$, the runway throughput during peak times is reduced, and the takeoffs of some flights are shifted to the next time window because of excessive pushback delays.

By putting the runway throughput data from Figure 4-8 and the numbers of taxiing aircraft from Figure 4-6 together, we can observe their relationship under the N-Control approach, as illustrated in Figure 4-9. In the NoGH case, the departure throughput saturates when the number of taxiing aircraft exceeds a specific threshold. When the departures are controlled, however, the distribution of data points looks different. In N-Control cases, the N values of the 15-minute periods are limited by

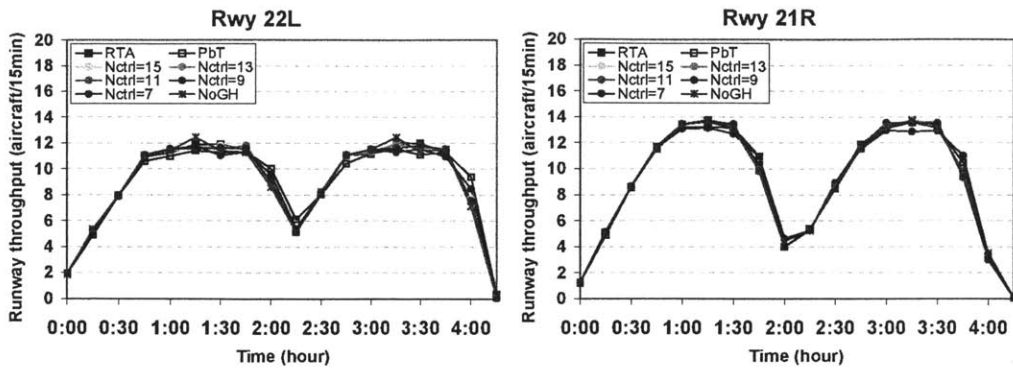


Figure 4-8: Runway throughput for both departure runways

the given queue capacity control parameter, while maintaining the maximum takeoff rate of each departure runway.

As the last performance measurement, the maximum takeoff delay values from the earliest possible takeoff times are compared in Figure 4-10. While the queue control parameter decreases in the N-Control cases, the largest takeoff delay in a 4-hr flight schedule generally increases because the gate-holding time becomes longer. When the N_{ctrl} value is low, there are large variations among flight schedule data sets, compared to the trajectory-based control approaches. Differently from the other metrics, however, we can find no clear tendency on the maximum takeoff delay.

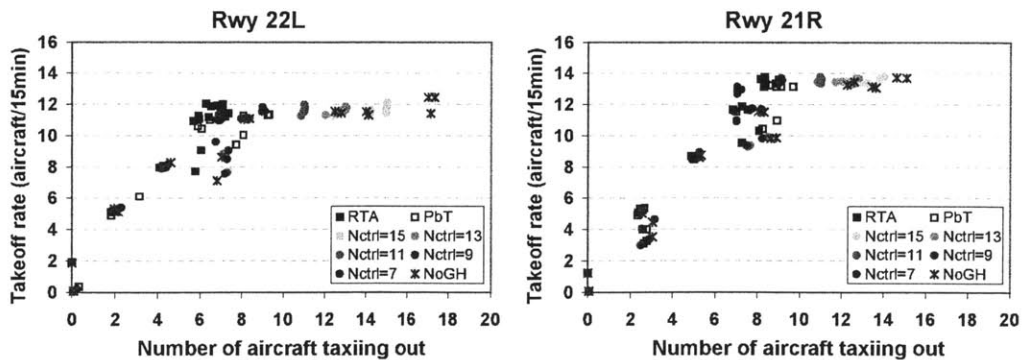


Figure 4-9: Runway throughput vs. number of taxiing aircraft

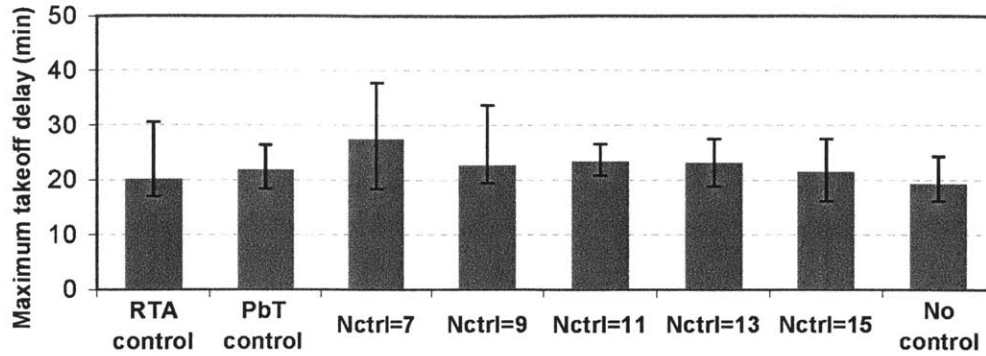


Figure 4-10: Maximum takeoff delay

4.5 Gate conflicts

For efficient departure management in airport operations, the gate-holding strategy can be used to mitigate the congestion on the surface and reduce the taxi-out time, while maintaining seamless takeoffs. However, when some departing flights are held at gates, there is a possibility that several arrivals cannot access their gates and have to wait for a while until the gates are cleared. This situation is called a *gate conflict*. A gate conflict usually happens at the gate that is allocated to an arriving flight, but still occupied by a departure. Depending on the airport layout, the gate conflicts may occur on the ramp area like a horseshoe shaped terminal having only one access point for multiple gates. Gate conflicts are also sensitive to gate characteristics such as airline's gate ownership, gate equipment and aircraft type limitation, which can constrain the number of usable gates. For example, only a few gates with dual-height jetway capability can handle A380 aircraft at many major international airports.

Minimizing the frequency that the gate conflicts occur is one of the main concerns in the gate assignment problem. Robust gate assignment algorithms have been developed to minimize gate conflicts and maximize the time gap between two consecutive flights using the same gate. These algorithms were tested at several busy airports [78,79,88,126].

In the N-Control approach, some departing flights can be held at gates for a long time, resulting in gate conflicts with arrivals. According to the recent study at New

York La Guardia Airport (LGA), the gate conflict frequency can be increased by 12 times when gate holding is on [79]. Other studies utilizing the gate-holding strategy also admitted the possibility of the gate-use conflicts [75,86,116]. In the discussion about the comparison of departure control approaches, it is therefore necessary to check the frequency of the gate conflicts in a given flight schedule and investigate their impacts on the arrivals. For this study, the flight schedules in Scenario 3 will be used continuously.

4.5.1 Gate conflict frequency

First of all, we count how many times gate conflicts occur. For a given flight schedule, the frequency of gate conflicts is measured by scanning the pushback times of departures and the estimated gate-in times of arrivals at each gate and then counting the situations that two flights share the same gate area simultaneously. The estimated gate-in times of arrivals used in the N-Control cases come from the NoGH case since the N-Control approach does not predict the movement of arrivals. Previous studies also showed that little difference on taxi-in delays was observed among the trajectory-based optimization cases.

It is assumed that the gate conflict occurs when the time gap between a departure and a next arrival assigned to the gate is less than the minimum round-trip time on the last taxiway link between the gate and the adjacent node on the ramp area. The minimum round-trip time includes the travel times required for the departure to move out and for the arrival to reach the gate. The sum of gate conflicts at the entire gates is calculated and averaged over 27 data sets in Scenario 3 flight schedules. For the trajectory-based control, both optimization approaches, represented by the FCFS and FPSr cases, are investigated. These are compared with the ‘No control’ case (NoGH case) and the N-Control cases having different queue control parameters.

The average gate conflict frequency over 320 arrivals in a 4-hr flight schedule for each control case is illustrated in Figure 4-11. In the N-Control cases, the gate conflict frequency increases as the queue capacity control parameter decreases. As long as the N_{ctrl} value is greater than 7, the frequency is lower than both optimization cases

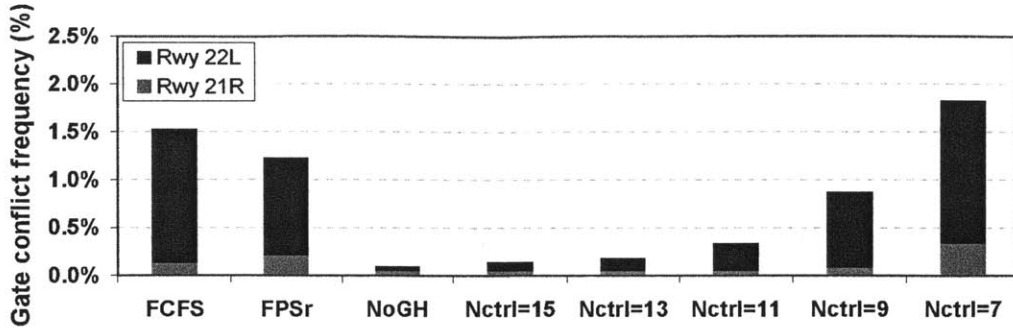


Figure 4-11: Gate conflict frequency

for the trajectory-based control, in which about 90% departures are held at gates. When $N_{ctrl} = 7$, however, the number of gate-held flights is half of the FCFS case, but more flights experience gate conflicts. As a reference, the gate conflict frequency at LGA was 2.45% in [79], although it is hard to compare the frequencies directly because the frequency is strongly dependent on airport layout, gate availability, and ramp control guidelines. It is also observed that the departures assigned to Rwy 22L experience more frequent gate conflicts than the flights going to Rwy 21R because longer gate-holding times are generally applied to the flights going toward Rwy 22L.

4.5.2 Options to resolve gate conflicts

A gate conflict situation rarely happens, but once it occurs, it produces adverse effects on the surface operations such as gate-in delays and additional workload of ramp controllers. The best way to prevent flights from undergoing possible conflicts around gates would be planning a plausible flight schedule that ensures the sufficient buffer time between two consecutive flights at the same gate in advance and implementing it as planned, while carefully monitoring off-nominal events on the ground. In real operations, however, gate conflicts are sometimes inevitable under congested conditions or at resource-limited airports.

When a gate conflict unavoidably happens, we can consider the following solutions from a tactical point of view.

Option 1: Arrivals wait for gate

The easiest ad hoc solution is to have the arriving flight wait on the ramp area until the departure moves out and the gate is cleared. This solution affects neither departure planning nor runway throughput, but directly increases the taxi-in times of arrivals. Although this option can lead to a gridlock situation to the following aircraft on the taxiway, we assume that the secondary effect can be neglected in this preliminary evaluation.

To study the effect of gate conflicts on arrivals, we calculate the arrival's gate-in delay, which is the difference between the taxi-in times with and without a pre-occupied departure at the gate. In the N-Control approach, the total gate-in delay value by gate conflicts dramatically increases when the queue capacity control parameter is lowered, as shown in Figure 4-12. In the trajectory-based optimization cases, additional taxi-in time by gate conflicts is 2.5-3.5 s/arrival or 3.4-3.8 min/conflict. Compared to the trajectory-based control, the taxi-in time added by introducing the N-Control method is smaller, as long as the N_{ctrl} value is greater than 7. In both control approaches, the total gate-in delays are dominated by the gate-held flights assigned to Rwy 22L. For the Solution 1, it seems reasonable that there is a proportional relationship between the gate-in delays by gate conflicts and the gate-holding times of departures for each runway. Note that the gate-in delays estimated in this study are only for the arrivals directly affected by gate holding and that the additional effects that can be propagated by taxiway blocking of the waiting aircraft are not considered. Therefore, the actual impact of this solution on the ground delay may be more significant.

Figure 4-12 shows that the departing aircraft assigned to Rwy 22L have higher values in both gate conflict frequency and total gate-in delays. According to Figure 4-13, the average waiting time of the arrival experiencing a gate conflict is also higher for Rwy 22L. As the N_{ctrl} value decreases, the gate-in delay per conflict induced by departures going to Rwy 22L increases significantly, whereas the Rwy 21R case shows invariant values.

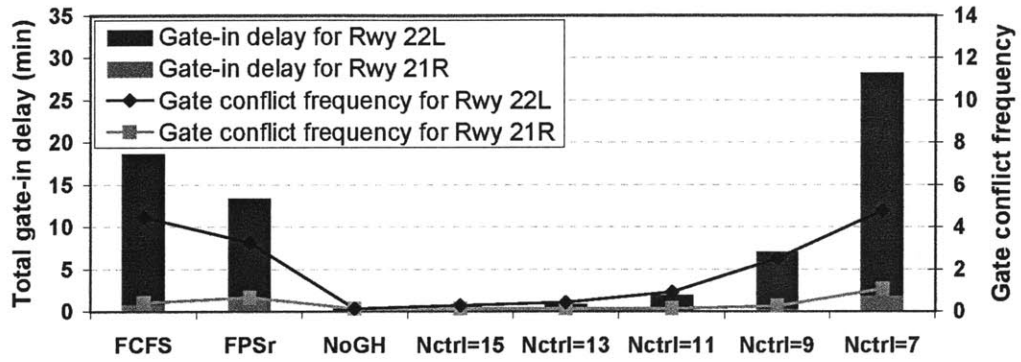


Figure 4-12: Total gate-in delays by departure runway

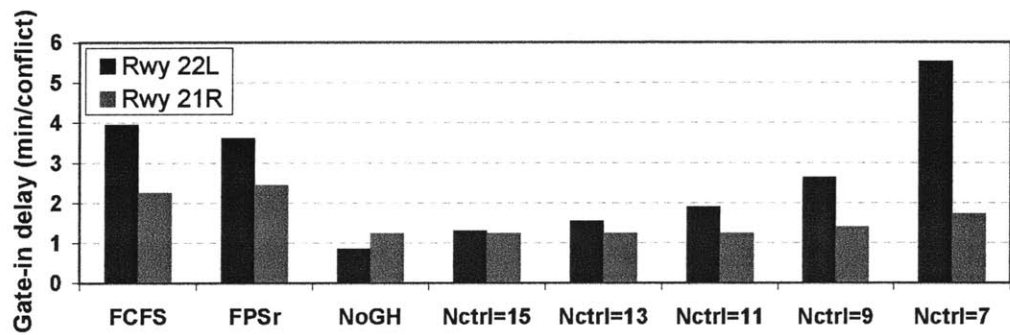


Figure 4-13: Gate-in delays by departure runway

Optimization case	FCFS	FPSr	NoGH	$N_{ctrl}=7$
Total gate-hold time reduction (min/dataset)	19.1	13.5	0.3	27.5
Total takeoff delay reduction (min/dataset)	5.4	0.1	0.0	15.4
Total taxi-out time increase (min/dataset)	13.7	13.4	0.0	12.1
Total taxi-in time reduction (min/dataset)	17.2	12.7	0.3	26.9

Table 4.4: Effects of early pushback of departures

Option 2: Departures pushback early

When the gate-holding strategy is applied, most gate conflicts are caused by holding departures at their gates for an excessively long time. In this case, it could be more effective to push the departures back earlier than the controlled pushback times. That is, when a gate conflict is expected, the departure is forced to leave its gate before the arrival allocated to the same gate reaches the access point to the terminal. This is based on the premise that predicting the gate-in times of arrivals is possible. This solution can minimize the effect on arrivals. However, the takeoff time schedule and sequence optimized by departure planning can be mixed up, resulting in increased taxi-out times and unintended takeoff delays. Furthermore, Solution 2 has a limitation on its implementation. If an early arrival raises a gate conflict, it may be impossible to release the departure flight from the impacted gate earlier than the scheduled pushback time.

Table 4.4 shows the changes from the Solution 1’s simulation results when Solution 2 is applied instead. As expected, we can save the taxi-in times of arrivals by pushing some departures back at gates earlier when gate conflicts are predicted. Reducing the gate-holding times of departures increases the taxi-out times and the taxiway congestion level because the departures leaving the gates early join the departure runway queues at unplanned times and interrupt the optimal takeoff sequence. However, the total takeoff delays are slightly reduced with the earlier pushback times.

Option 3: Gate reassignment

If there are enough available gates to accommodate the traffic demand at the airport, finding an alternative gate can be the easier solution for a ramp controller to resolve

the gate conflict problem. However, when assigning an arrival to an alternative gate, the controller needs to account for operational constraints like airlines, aircraft type, and gate equipment. In order to avoid passenger confusion, gate changes are generally made in advance. Robust gate assignment to minimize the occurrences of gate conflicts is an ongoing topic of research in airport operations [88,126], but it is beyond the scope of this thesis.

4.6 Conclusions

In this chapter, we described two different control approaches to managing the surface traffic efficiently. The aggregate queue-based departure control approach, represented by N-Control, was introduced first. This approach was then compared with the individual aircraft trajectory-based control approach, called RTA control. The PbT control was also compared as an intermediate control approach before the advanced technologies required for the RTA control are realized.

These three methods were implemented for high traffic flight schedule scenarios at DTW and evaluated with respect to various airport performance metrics. The simulation results of the N-Control method showed that as the queue capacity control parameter became lower, the better taxiway performance could be achieved at the expense of other metrics such as takeoff delays and gate conflicts, while maintaining the same level of runway throughput. The other control approaches based on the aircraft trajectories showed more taxi-out time savings by applying the gate-holding strategy more aggressively. If implementable, the RTA control would provide us with the most significant taxi time savings during peak times without losing any runway performance.

It was also found that the departure control by gate holding could lead to gate conflicts in real operations. A few options to resolve the gate conflict situations were reviewed in this chapter. The waiting of arrivals would be an easy way for ramp controllers to implement, but produce the gate-in delays. According to the simulation results, the earlier pushback of departures could be a better solution to minimize the

side effect of gate conflicts.

Chapter 5

Impact of uncertainty on airport performance and planning

5.1 Uncertainty in airport operations

All of the optimization and simulations done so far have been based on the assumption that airport operations are deterministic. That is, the operational parameters used in both optimization and simulations were fixed, and the disturbances from them were not allowed. However, there exist lots of variations and uncertainties in real operations. They include actual pushback times of departures, actual landing times of arrivals, taxi speed of aircraft, holding time on the taxiway and in departure queue, roll distance of takeoff or landing, and actual separation times between consecutive flights over the runway. These uncertainties are caused by the variety of airline procedures, aircraft types, pilots, air traffic controllers, surface conditions, weather, and operational environment. Since these uncertainties cannot be neglected, it is necessary to reflect them in the simulations and investigate the impacts of them on ground delays, runway throughput, and any other airport performance metrics. However, some less frequent uncertainty factors that lead to large perturbations from the original plan, such as airport configuration changes, aircraft malfunctions and emergency situations, are beyond the scope of this thesis.

SIMMOD provides a stochastic process function in order to accommodate the

presence of uncertainties in airport operations. This simulation tool can model several uncertain factors in the form of random variables and produce statistical outputs representing the variations in air traffic phenomena through many iterations. The random number streams provided by SIMMOD include gate selection and occupancy time, multiple arrival and departure, takeoff and landing roll distance, inter-aircraft separation, flight lateness, bank late flight holding probability, bank late transfer time, and arrival and departure clone [2,7].

In this chapter, several uncertainty factors in airport operations are modeled as random variables at microscopic level in a fast-time computer simulation tool (SIMMOD) and studied through stochastic simulations in order to investigate their impacts on ground delays and taxi times. These uncertainties consist of:

1. Actual pushback times of departures, which are random perturbations of the given flight schedule using gate service (occupancy) times in SIMMOD,
2. Varying taxiway entrance times of arrivals, which can be varied by landing roll distances in SIMMOD,
3. Different taxi speeds on the taxiway and the ramp areas depending on the flights,
4. Uniformly distributed separation times between takeoffs, determined by the in-trail separation multiplier in SIMMOD.

In the next section, we describe the simulation modeling approach and the flight schedule scenarios used in the simulation-based uncertainty studies.

5.2 Simulation modeling for uncertainty studies

5.2.1 Simulation model validation

To run the various simulation scenarios for uncertainty studies in surface operations, a basic SIMMOD simulation model for an airport should be developed first. To make the process easy, DTW airport is chosen again for these uncertainty studies. So, the

same SIMMOD model constructed in Chapter 2 is used for the uncertainty studies as it is. The operational parameters defined in the model also remain the same, except for the random variables varied in the uncertainty experiments.

As mentioned earlier, several parameters in the SIMMOD input data can be varied in the stochastic process using random number seeds. SIMMOD can repeat the entire surface traffic movements with the given input and with randomly generated parameters as many times as needed. After iterating simulation runs, SIMMOD brings out the statistics from the simulation result, such as aircraft movement starting and ending times for individual flights, average travel time and delay on the ground and in the airspace, and fuel consumption (if available).

Before using this SIMMOD model for various uncertainty studies, the simulation model is validated with actual flight data from DTW airport on 8/1/2007. The whole day flight schedule is simulated in SIMMOD under the same operating conditions, and the resultant travel times on the surface, including the unimpeded taxi time and ground delay for each flight, are analyzed in this section.

Figure 5-1 shows the average taxi-out times from the SIMMOD simulation and the surveillance data for each 5-min interval during the day. The two curves are similar for low traffic levels, like the early morning periods. At most times, however, there is a significant gap between the simulated and actual values. As the number of flights moving on the ground increases, the gap becomes larger. The reason of this gap is that no uncertainty is taken into account in this SIMMOD simulation. As a result, the deterministic simulation neglects the effect of uncertainty on travel times and the increased interactions between flights in high traffic density situations. Other causes for the difference between the simulated and observed values may include measurement errors and missing records in the surveillance data.

To validate simulation parameters such as taxi speeds and routes, the unimpeded taxi-out times from the SIMMOD simulation and a queueing model are also compared. The unimpeded taxi times for departures can be estimated by the linear regression method used in the queueing model for airport departure process using historical traffic data [114]. This method was validated for several major airports [115] and applied

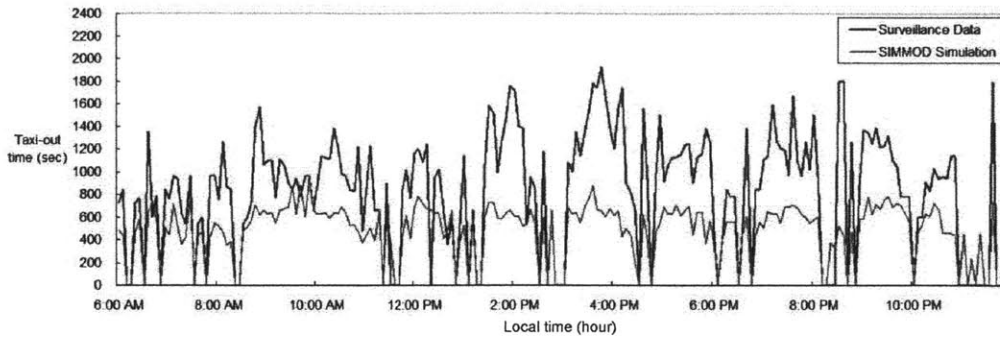


Figure 5-1: Taxi-out time comparison between SIMMOD simulations and surveillance data

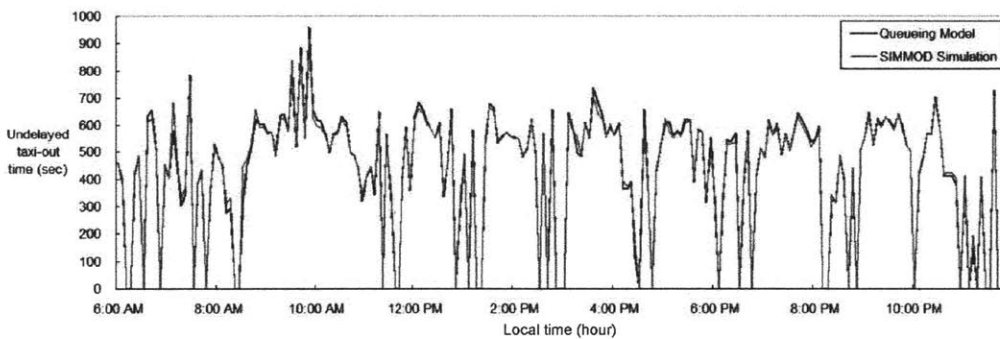


Figure 5-2: Unimpeded taxi-out time comparison between SIMMOD simulation and queuing model

to DTW airport, considering airlines, gates, runways, and weather conditions [87]. As shown in Figure 5-2, the unimpeded taxi-out time curves from the queuing model and the SIMMOD simulation are very similar.

The result from Figures 5-1 and 5-2 implicates that the SIMMOD airfield model and the related simulation parameters have been modeled appropriately and that the gap between real operations and simulation can be diminished by considering the uncertainty in surface operations, of which the effects on the taxi time increase as the traffic becomes congested, in the simulation.

5.2.2 Flight schedules used in simulations

To evaluate the impacts of various uncertainty factors in surface operations in more detail, we use two flight schedule scenarios at DTW, depending on the traffic level.

The first schedule scenario representing the current traffic level is one hour of data at DTW between 7:00PM and 8:00PM on August 1st, 2007, which is one of the busiest times on that day. In this scenario, two different flight schedules are simulated and compared for each uncertainty effect experiment. One corresponds to the originally scheduled pushback times, and the other schedule corresponds to the optimized pushback times obtained by holding flights at their gates (gate-holding strategy). The optimized flight schedule is from the 2-CPSd case using the three-step approach. The pushback times are optimized so that their takeoff times are appropriately separated, and so that the ground delays and the sizes of departure queues are minimized. Through the stochastic simulation, we will check if these benefits from optimization are still valid under the uncertain operational conditions. Arrivals are also simulated to create a realistic surface environment.

The second flight schedule scenario is from Scenario 3 in Chapter 3, which represents the higher traffic level at DTW in the future. This scenario models four-hour flight schedules having two peaks, of which the hourly demand rate is double the current traffic level. In this scenario, we also test two kinds of flight schedules, which are original and optimized departure schedules. To obtain the optimized flight schedule, the FPSr case based on the integrated approach is used. By comparing the stochastic simulation results from the two schedule scenarios, we can investigate the relationship between traffic level and uncertainty effects on the airport surface.

5.2.3 Iterations with random variables

Parameters in the SIMMOD input data can be varied using random seeds. Given the flight schedule and the fixed operational inputs, SIMMOD can repeat the entire surface traffic movements with randomly generated parameters as many times as needed. The random parameters varied for the uncertainty studies in this chapter include gate occupancy time for pushback time perturbation, landing roll distance for runway exit time perturbation, and in-trail separation multiplier for inter-departure time perturbation. In case of taxi speed perturbation, aircraft taxi speeds are randomly distributed within a given range by using a separate pre-processing module.

After iterating simulation runs, SIMMOD will compile statistics from the simulation results, such as movement start and end times of aircraft at gates and runways, travel times and delays on the ground and in the airspace.

5.3 Pushback time variations

Airport taxiway optimization models generally assume that flights leave the gates exactly at the optimized pushback times. In reality, however, uncertainties in the pushback process make it very unlikely for an aircraft to meet its assigned pushback time. A flight may move out from the gate later than the scheduled pushback time due to late passengers, delayed loading of galley carts for cabin service, unexpected maintenance checks, waiting for clearance from the control tower, or communication with ground crews. Similarly, a flight may depart earlier if there are no delays or disruptions during the pushback process.

This uncertainty in pushback time can be modeled in SIMMOD by using randomized gate service times within a given range. For example, if flight *A* is scheduled to depart at 9:00AM and the mean gate service time is 30 minutes, then flight *A* will show up at 8:30AM in SIMMOD simulation in the absence of uncertainty. If we allow ± 5 min deviation from the mean value of gate occupancy times, the actual pushback time will be chosen as a random value between 8:55AM and 9:05AM. Each flight in the flight schedule has a different deviation independently drawn from the given distribution.

5.3.1 Pushback time uncertainty study at current traffic level

First, the random variables for gate occupancy times are applied to the 1-hr flight schedule scenario with the current traffic level described in the previous section. The pushback time deviation from the deterministic flight schedule is assumed to range between 0 min (no uncertainty) and 5 min. In this case study, a truncated Gaussian distribution and a uniform distribution are considered for the pushback time uncertainty. For each probability distribution, 100 different flight schedules are generated

Flight Schedule	Probability Distribution	Total Ground Delay / Simulation Run (min)			
		21R_Dep	22L_Dep	22R_Arr	27L_Arr
Initial Pushback Time	Deterministic	5	49	1	1
	Gaussian	8	54	1	0
	Uniform	9	56	1	0
Optimized Pushback Time	Deterministic	1	15	1	0
	Gaussian	8	32	1	0
	Uniform	9	36	1	0

Table 5.1: Impact of pushback time uncertainty on the ground delay

and simulated in SIMMOD for both the initial and optimized pushback schedules.

Table 5.1 summarizes the total ground delay per iteration in minutes categorized by runway (and averaged over 100 trials). For each pushback schedule, the deterministic case is used as a baseline. We see that the ground delay increases for both departure runways 21R and 22L, as the uncertainty of pushback times increases from deterministic to Gaussian, and then to a uniform distribution. By contrast, there is little effect on arrivals because the landing schedules remain deterministic. The simulations also demonstrate the benefits of the gate-holding policy. For the departure runway 22L, the optimized pushback schedule has a lower ground delay even with uniformly distributed pushback uncertainty compared to the deterministic case with no gate-holds. This result suggests that the solutions recommended by deterministic surface traffic optimization provide benefits even in the presence of pushback time uncertainty.

This experiment can be extended to observe the effect of the deviation limit on the ground delay. The actual pushback time of a flight may sometimes be beyond the 5 min deviation from the given schedule. So, the same simulations have been implemented with various deviation limits from the deterministic pushback times, ranging from 0 to 15 minutes. In terms of flight operations, however, it is not allowed that a flight leaves the gate more than 5 minutes earlier than the schedule in the simulations. For the same 1-hr flight schedule data, 100 different samples, in which the pushback times of departures are randomly selected over the given range using a uniform distribution, are generated and run in SIMMOD for each pushback time deviation limit.

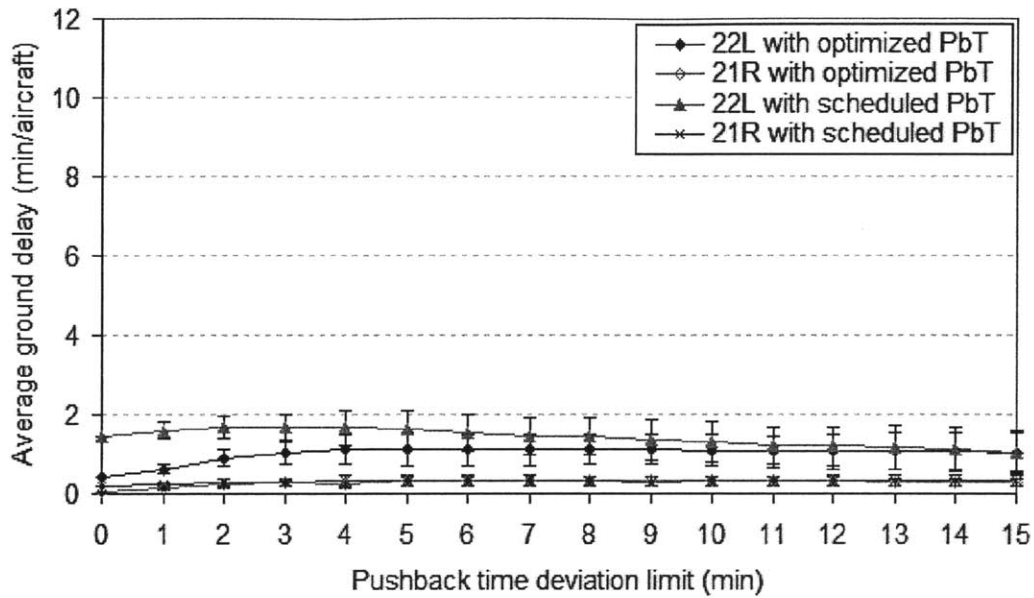


Figure 5-3: Average ground delay per departure with pushback time perturbation at current traffic level

Figure 5-3 presents the average ground delay per departure as a function of the maximum pushback time deviation. The whiskers in the graph denote the standard deviation of the delays from the 100 simulations. For both departure runways, the ground delays for both original and optimized pushback time schedules increase until the range of the pushback time perturbation becomes ± 5 min from the given schedule. For the scheduled pushback time case, the ground delay of a departure going to Runway 22L decreases gradually when the allowed deviation increases beyond 5 minutes. This reduction shows that the unintended pushback delays may act as an implicit gate-holding policy when the departure traffic demand is high, by keeping aircraft at their gates until the surface congestion has decreased.

5.3.2 Pushback time uncertainty study at higher traffic level

Next, the same uncertainty study about pushback time perturbation is performed with the second traffic scenario, 4-hr flight schedules having 2 peak periods used in Chapter 3. As in the previous experiment, 100 different trials having various push-

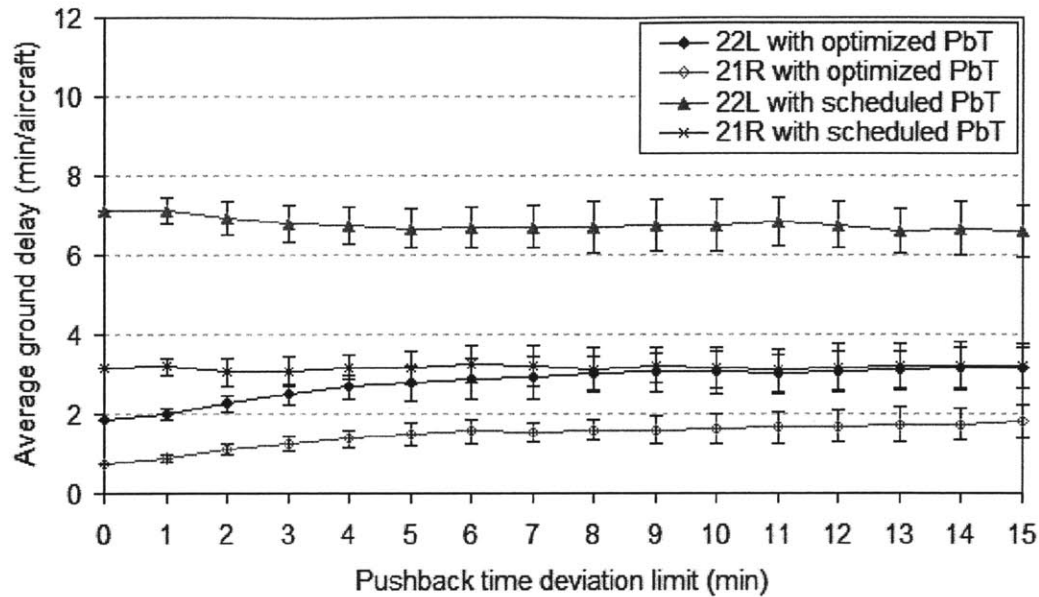


Figure 5-4: Average ground delay per departure with pushback time perturbation at higher traffic level

back time deviations are simulated in SIMMOD. The pushback times are uniformly distributed within a given deviation range, and the deviation limit applied to the simulations increases from 0 to 15 minutes.

Figure 5-4 shows the average ground delay per departure depending on the uncertainty level of the pushback times. The curves are categorized by the assigned departure runway and the schedule optimization using the gate-holding strategy. According to the simulation results, the pushback time uncertainty makes a limited effect on the ground delay. When the pushback times of departures are optimized, the ground delay increases as the pushback time perturbation range becomes wider, but the delay curves saturate at around 8 minutes of the deviation limit. On the other hand, when the departures leave their gates at the scheduled pushback times, their average taxi-out delay does not change much along with the uncertainty level in pushback times. It seems that the adverse effects on the ground congestion by uncertain pushback times are offset by gate-holding caused by the uncertainty.

We now investigate how the traffic demand level affects the stochastic simulation results showing the relation between the pushback time uncertainty and the taxi-out

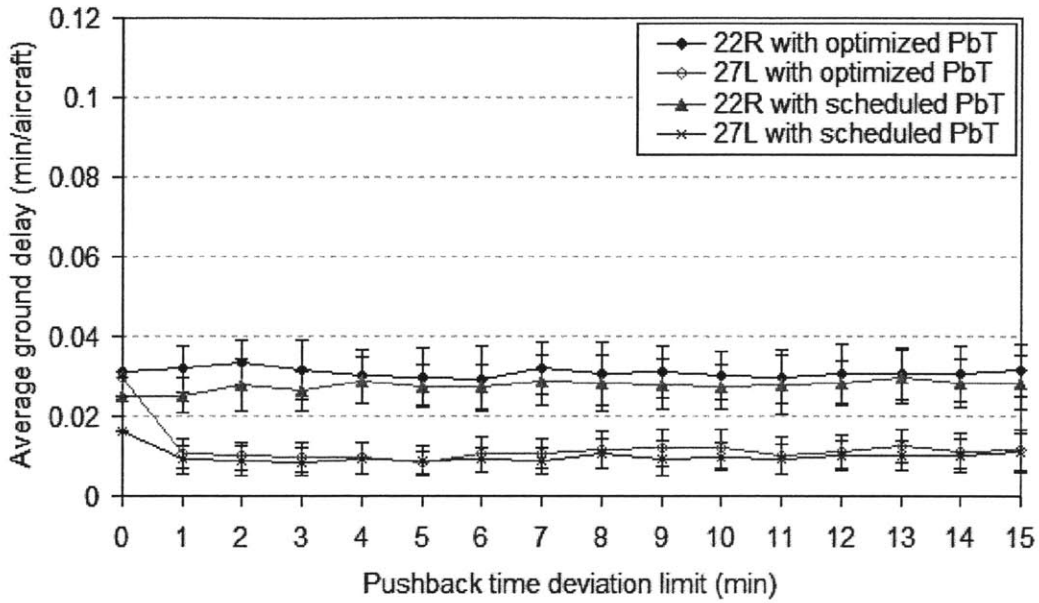


Figure 5-5: Average ground delay per arrival with pushback time perturbation at higher traffic level

delay. By comparing Figures 5-3 and 5-4, we can observe that the absolute values on ground delays (and the taxi-out time savings by pushback time optimization) are always much higher when the traffic is more congested. However, the two graphs commonly show that the taxi-out delay curves saturate at some points as the maximum deviation from the given pushback times increases, regardless of the surface traffic level. Also, the benefits from the gate-holding strategy on the ground delay are still valid in the presence of the pushback time uncertainty for both current and higher traffic levels.

For arrivals, there are little impacts on their average ground delays. As shown in Figure 5-5, the pushback time perturbation applied to departing flights does not affect arrivals' ground movements, unless gate conflicts occur. Note that the scale on the y-axis in Figure 5-5 is a hundredth of Figure 5-4's.

5.3.3 Impacts of pushback time uncertainty on gate conflicts

The pushback time uncertainty can lead to gate conflicts. If a departing flight actually leaves its gate later than the given pushback time for some reason, an arrival assigned to the same or adjacent gate may have to wait on the ramp area to avoid a conflict with the departure. Therefore, we need to investigate how often the gate conflicts occur in the presence of pushback time uncertainty.

The gate conflict frequency can be obtained by analyzing the gate arrival and departure times from SIMMOD simulations. For this analysis, the simulation results for the higher traffic level scenario are used because gate conflicts seldom happen in the current traffic level flight schedules. As performed in the previous section, 100 simulations are implemented for each pushback time deviation limit ranging from 0 to 15 minutes, and each simulation has randomly deviated pushback time schedules within the given perturbation limit. The same simulations are run with both initially scheduled and optimized pushback times.

Figure 5-6 illustrates the averaged gate conflict frequencies of the original and optimized flight schedules as a function of pushback time uncertainty level. According to the graph, it seems that the gate conflicts can happen due to the pushback time uncertainty, but the frequency does not proportionally increase as the uncertainty range becomes wider. Note that the pushback delay of a specific departure flight is not directly related to the deviation limit of the pushback time perturbation, but randomly determined within the given deviation limit. Compared to the original departure schedule, the optimized schedule is more likely to have gate conflicts in the presence of pushback time uncertainty because most departures are held at their gates. This prediction can be confirmed by observing Figure 5-6, showing that gate conflicts occur more frequently (up to 3%) when the pushback times are optimized.

5.4 Landing time variations

Arrivals also have uncertainty associated with their runway exit times, that is, the times when they enter the taxiway system. This uncertainty can be modeled by vary-

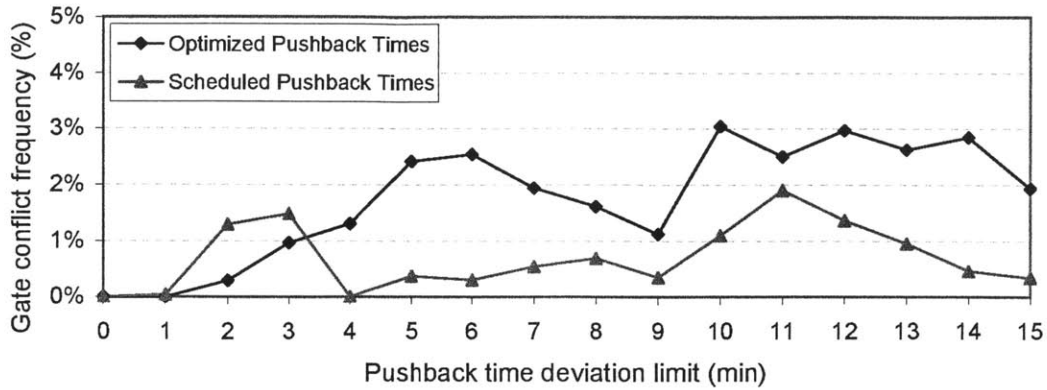


Figure 5-6: Gate conflict frequency by pushback time uncertainty at higher traffic level

ing the landing-roll distances in SIMMOD. The landing roll distance is affected by many factors, such as aircraft weight, approach speed, braking performance, headwind, runway surface condition, slope, and human factors [66]. In this uncertainty study, the deviation range from the normal landing roll distance used in SIMMOD is set to ± 500 ft. It is also assumed that arrivals use the same runway exit so as to restrict the uncertainty to the runway exit time alone, and that the landing sequence does not change due to this uncertainty.

For the same flight data as the previous case studies, 100 trials were implemented by using random seeds in SIMMOD. The Monte Carlo simulation showed no effect on the ground delay. The perturbation in roll distance impacted just the gate-in time of each arrival. This result is reasonable since the inter-arrival times for safe landing provide consecutive flights with sufficient spacing on the taxiway.

5.5 Taxi speed variations

The objective of the case studies described in this section is to investigate the impact of flights moving at differing taxi speeds on airport surface traffic. In the previous simulations, it was assumed that all flights taxiing on the ground move at the same taxi speed, which is the average value of various taxi speeds observed at the airport. With this assumption, the trailing flight on a taxiway can keep a constant separation

distance from the leading flight on the same taxi route. In practice, however, taxi speeds may differ from flight to flight [55] due to many factors such as aircraft type, pilot behavior, operational procedures, taxiway length, etc. We therefore simulate differing taxi speeds and scrutinize the impacts of taxi speed variations on the ground delay.

5.5.1 Case study 1: Taxi speed variations on taxiways only

First, we assume that the pushback times for departures and the runway exit times for arrivals are known and deterministic. Each flight is assumed to have a different taxi speed within a given range on the taxiway area, which it maintains along the entire taxi route. The upper and lower bounds of the taxi speed range would be determined by aircraft performance, ground congestion, and operational rules at the airport. In this case study, all flights are assumed to maintain a constant speed of 7 knots in the ramp areas. We also suppose that there are no significant differences in taxi speeds based on aircraft type or between arrivals and departures.

The same flight schedule scenarios at DTW as the previous uncertainty studies are used. Values within the given taxi speed range are randomly generated using a uniform distribution, and assigned to the flights in the schedule. The mean value of the taxi speeds is set to 18 knots, which is consistent with the parameters used in the optimization models for aircraft taxiway scheduling.

For the Monte Carlo simulation, 100 trials with randomly generated taxi speeds are run by SIMMOD for each taxi speed range condition. The data sets contain the same flight schedules, pushback times and landing times, but different taxi speeds are assigned to the same flight in each trial. To investigate how the uncertainty in taxi speed affects the ground delay, six different taxi speed deviation ranges from 0 to ± 5 knots are studied. Also, for comparison, the simulations are conducted with the original schedule and the optimized pushback time schedule.

Figure 5-7 illustrates the average ground delays per flight with standard deviations from 100 simulations implemented for the current traffic level scenario along with taxi speed perturbation, depending on the runway and on the taxi speed range. In these

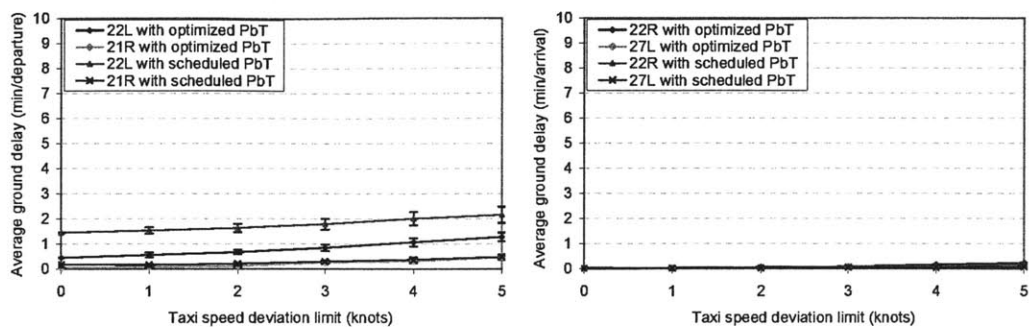


Figure 5-7: Average ground delay with taxi speed perturbation on taxiway area at current traffic level

simulation results, the ground delay includes holding for runway crossings, intersection holds to avoid conflicts, wait times in the departure queue, and holds for maintaining separation on the taxiway due to a slower flight ahead. In Figure 5-7, the average ground delay for departures increases as the taxi speed range becomes wider. This tendency is to be expected because the taxi speeds of flights are constrained by slower flights. For instance, if a leading flight is slower than a group of flights behind it along the same taxi route, the trailing flights cannot taxi faster than the leading one, resulting in increased taxi times and wait times in the departure queue. In rare cases, the ground delay can decrease with increased taxi speed perturbation because the different taxi speeds may widen the spaces between flights on taxiway and runway. In general, however, the average ground delay increases as the taxi speed deviation limit increases.

By contrast, there is little impact on the ground delay of arrivals since the average delay per arrival from each runway is less than 15 seconds for any of the taxi speed ranges. This result is due to the fact that arrivals are already separated enough when exiting the runway, resulting in minimal interactions with the following aircraft. Taxi routes for arrivals are almost independent of the paths for departures, except in the ramp area, further minimizing interactions between them.

In the simulation results, the original and optimized pushback time schedules are also compared. For the more congested runway (Runway 22L), the ground delay of the optimized pushback time schedule is always much less than the delay of the

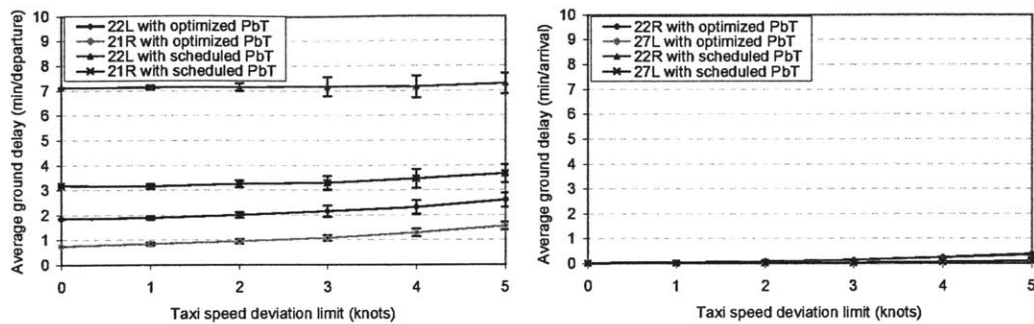


Figure 5-8: Average ground delay with taxi speed perturbation on taxiway area at higher traffic level

original schedule. The ground delay values for the two schedules are similar for the less congested runway (Runway 21R), and they become more similar as the taxi speed deviation increases. The additional ground delay compared to the deterministic case is greater for the optimized schedule case as the taxi speed deviation increases, possibly because the optimized flight schedule is more sensitive to taxi speed uncertainty.

Figure 5-8 shows the average ground delay curves as a function of taxi speed perturbation range for the higher traffic level scenario used in Section 5.3.2. The simulation results are fundamentally similar to the current traffic level scenario results shown in Figure 5-7, although the initial ground delay values in the constant taxi speed case are different depending on the surface traffic level. When the taxi speed variation among aircraft moving on the taxiway area increases, the average ground delays also gradually increase, but their growth rates are not steep for both departures and arrivals.

5.5.2 Case study 2: Taxi speed variations on taxiway and ramp areas

In the previous case study, different taxi speeds were applied to the taxiway areas only. This uncertainty can be extended to the ramp area where aircraft move from/to gates around the terminals at slower speed. When the flights move both on the ramp and taxiway areas at differing speeds, ground delay is expected to increase because

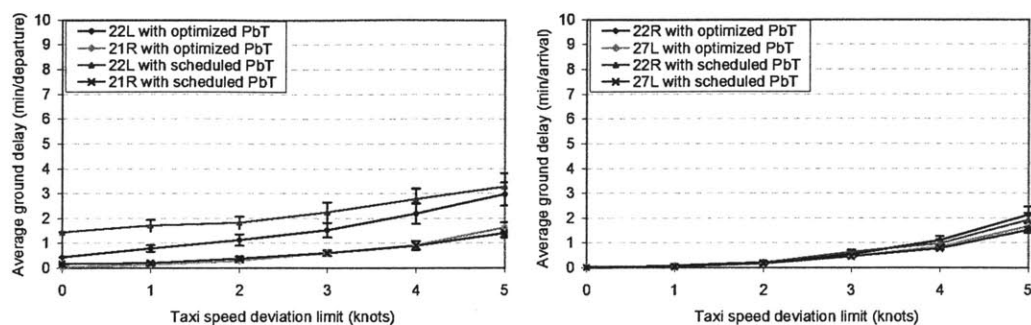


Figure 5-9: Average ground delay with taxi speed perturbation on taxiway and ramp area at current traffic level

of more frequent interactions, especially in the ramp area.

As in the previous case, taxi speed values perturbed around the average taxi speed are assigned to flights. The same deviation is applied to the taxi speed values on both ramp area and taxiways. For example, if the random deviation of a flight is +1.5 knots, the assigned speed will be 19.5 knots on the taxiway and 8.5 knots in the ramp area. While increasing the taxi speed deviation limit from 0 to 5 knots, SIMMOD runs 100 simulations with different taxi speed distributions for each taxi speed range. This uncertainty study is also performed with respect to the same flight schedules used in the previous case study.

Figure 5-9 shows the average ground delay from each runway for the current traffic level scenario. As expected, the absolute values of the ground delay are significantly increased for all cases, compared to Figure 5-7. As in the previous case study, the delay increases as the taxi speed range increases. In contrast to the previous results, arrivals also experience increased ground delay due to taxi speed perturbation. This fact is due to arrivals sharing the ramp area with departures. It is worth noting that most arrival ground delays in the SIMMOD simulations occur in the ramp area. We also note that the optimized pushback time schedule shows one minute less delay per departure for Runway 22L compared to the original schedule, but the difference becomes narrow as the taxi speed variance increases. On the other hand, flights using other runways show almost the same delay values for all taxi speed ranges, regardless of the pushback time optimization.

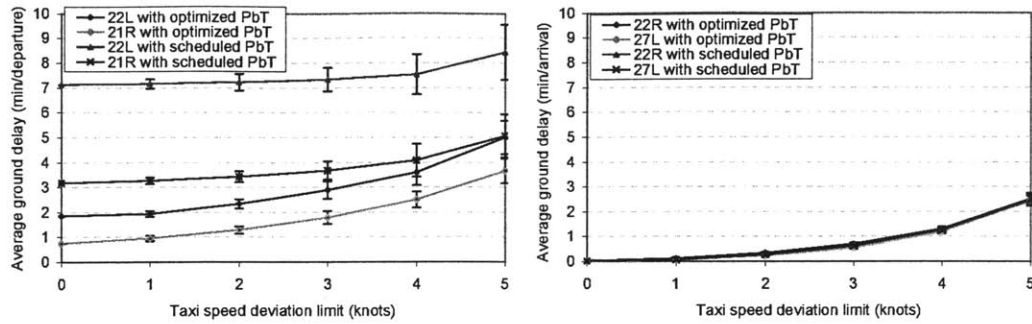


Figure 5-10: Average ground delay with taxi speed perturbation on taxiway and ramp area at higher traffic level

The average ground delay curves for the higher traffic level scenario are drawn in Figure 5-10. For the departures in the more congested surface traffic condition, the initial ground delays with no taxi speed perturbation are much higher than the delay values in Figure 5-9. However, it seems that the rate of delay increase to the taxi speed perturbation is similar, regardless of the traffic level. For both departure runways, the optimized schedule is more sensitive to the taxi speed perturbation, but always shows much lower taxi-out delays in the presence of taxi speed uncertainty. The average ground delay of arrivals also increases significantly when both taxiway and ramp areas are affected by taxi speed perturbation. Similar to the departures, however, the growth rate of delays looks uncorrelated with the traffic congestion level.

5.5.3 Case study 3: Taxi speed variations on taxiway and ramp areas, with faster arrivals

The prior case studies assumed that there was no difference between arrivals and departures in terms of taxi speeds. However, arrivals tend to taxi faster in practice. Analysis of surface surveillance data at DFW has shown that arrivals are about 2 knots faster than departures while taxiing [55]. While the average taxi speed was assumed to be 18 knots in the previous case studies, the mean values of taxi speeds in this case study are assumed to be 17 knots for departures and 19 knots for arrivals on the taxiway. On the ramp area, however, it is assumed that both departures and

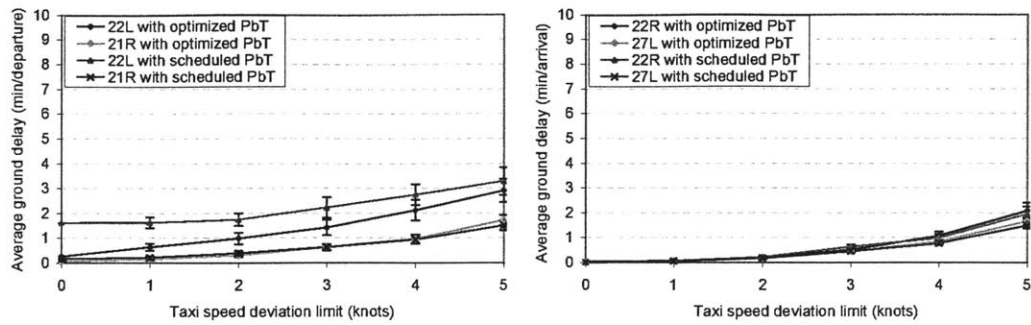


Figure 5-11: Average ground delay with taxi speed perturbation on taxiway and ramp area with faster arrivals at current traffic level

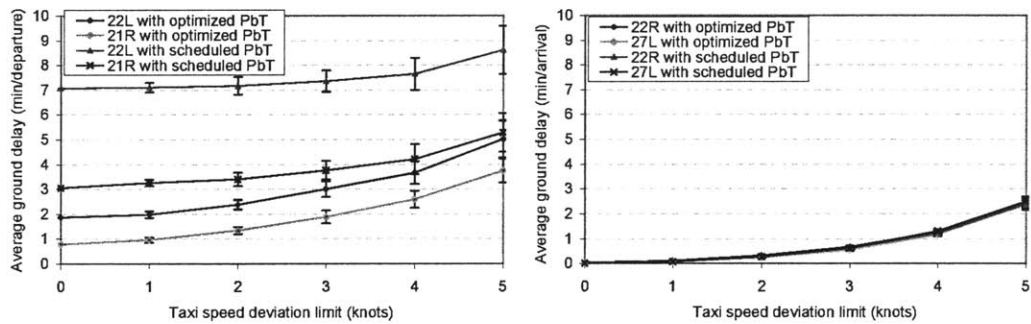


Figure 5-12: Average ground delay with taxi speed perturbation on taxiway and ramp area with faster arrivals at higher traffic level

arrivals have the same average of 7 knots.

The resultant average ground delays for this case study are shown in Figures 5-11 and 5-12 for two flight schedule scenarios having different traffic levels. The average ground delay variation with taxi speed deviation in Figure 5-11 is almost the same as in Figure 5-9, suggesting that ground delay is not affected by the absolute values of the average taxi speed, but only by the deviation range. The average taxi speed value in the stochastic simulation affects the unimpeded taxi time alone. The same conclusion can be reached by comparing the ground delay curves from Figures 5-10 and 5-12 for the higher traffic level scenario.

5.6 Inter-departure time variations

The takeoff separation times between consecutive departures differ in real operations, even when the weight classes are the same. The inter-departure separation time variation is modeled in SIMMOD using a random variable for the in-trail separation multiplier. This factor multiplies the minimum separation time to yield the separation time in the simulation. In this analysis, various separation times between takeoffs are randomly generated in SIMMOD within the given range, while the minimum separation requirements are maintained. The upper limit of the in-trail separation multiplier applied in this experiment ranges from 1.0 (tight separation) to 1.5 (conservative operation). For each case, 100 simulation runs are performed.

For the current traffic level scenario, Figure 5-13 illustrates the average simulated ground delay along with the upper limit of the separation time multiplier, depending on the runway and on the flight schedule. As expected, the ground delay linearly increases as the range of separation times widens. The increased delay on the ground comes from the increased wait time in the departure queue because the following flight needs to wait longer before takeoff. Furthermore, the waiting time is propagated when the departure queue is full. For Runway 22L in the graph, the average ground delay for the optimized pushback time case reaches 1.4 minutes per departure when the multiplier limit is 1.4, implying that conservative runway operations can cancel out the benefits from taxiway schedule optimization. It is also evident that the separation time uncertainty can decrease runway throughput.

Figure 5-14 shows the average ground delay as a function of the inter-departure separation time uncertainty for the higher traffic demand. In this traffic scenario, the absolute ground delay values increase due to separation requirements, even in no uncertainty cases. Because of the delay propagation in departure queues, the ground delays for the higher traffic level scenario are more sensitive to the inter-departure time uncertainty, but the benefits from surface traffic optimization are greater for both departure runways.

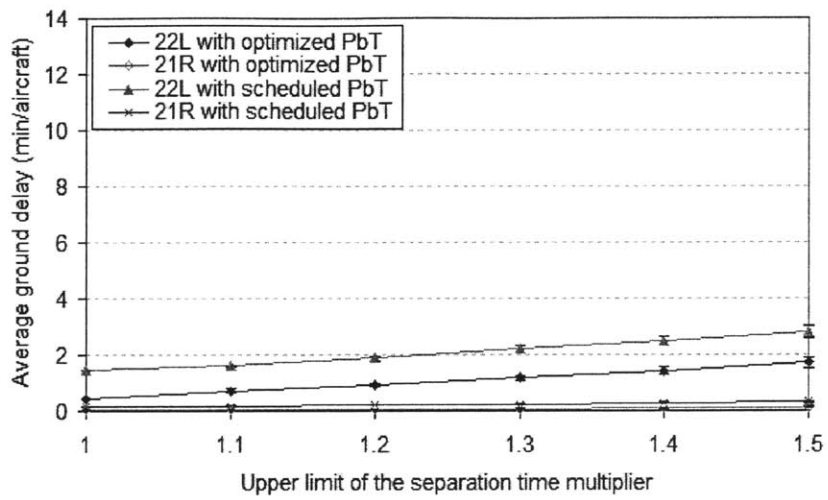


Figure 5-13: Average ground delay with separation time perturbation at current traffic level

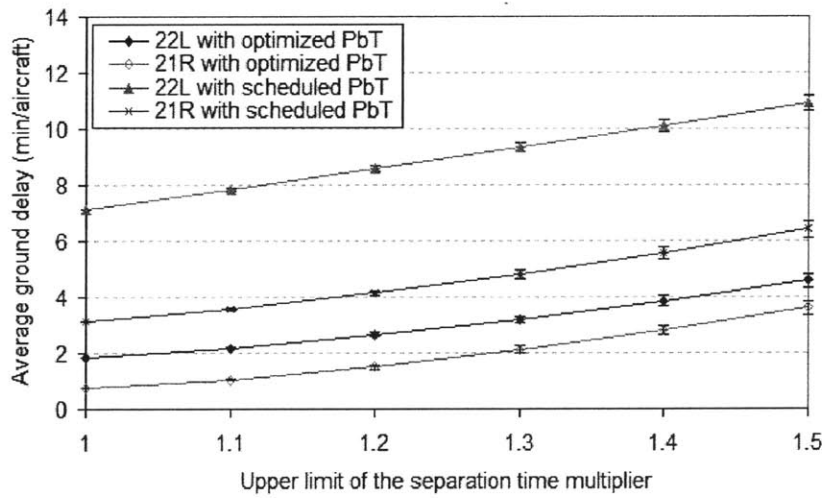


Figure 5-14: Average ground delay with separation time perturbation at higher traffic level

5.7 Conclusions

In this chapter, significant uncertainty factors in airport operations were identified, and stochastic simulations using SIMMOD were developed for evaluating the impacts of the uncertainties on airport performance. The simulation model was validated with actual flight schedules at DTW. Monte Carlo simulations for various uncertainty studies were then conducted with two flight schedule scenarios at the same airport, depending on the traffic level.

Simulation results showed that the ground delays saturated or decreased as the uncertainty in pushback times grew because the pushback delay due to the uncertainty acted like gate-holding. Uncertain runway exit times for arrivals did not significantly impact airport performance, apart from gate arrival times. It was also shown that perturbations in taxi speeds resulted in significant increases in ground delay for departures. By contrast, the taxi-in times of arrivals increased only when there were taxi speed variations in the ramp area, where arrivals interact with departures; however, the ground delay did not depend on the absolute value of the average taxi speed. In most cases, the delay growth rate to the taxi speed perturbation was also independent of the traffic level. Uncertainty in inter-departure separation times increased wait times in the departure queue, while reducing runway throughput.

The case studies presented in this chapter also compared simulations of originally scheduled and optimized pushback times and showed that the surface traffic optimization based on a deterministic model can still provide benefits in the presence of uncertainties.

Chapter 6

Summary and Next Steps

6.1 Summary of this thesis

In this thesis, we developed two different approaches to optimizing runway and taxiway schedules for efficient airport surface planning. The integrated approach optimized both runway and taxiway schedules simultaneously in a single MILP model, while the three-step approach sequentially combined a taxi-out time estimation module, a runway scheduling algorithm, and a MILP model for taxiway scheduling. These two approaches were compared using actual flight schedules at Detroit airport (DTW). The optimization results indicated that the three-step approach could provide better computational performance without a significant sacrifice in optimality. We also developed a fast-time simulation model for DTW using SIMMOD to model the air traffic flow on the surface and verify the benefits from the departure planning based on the surface traffic optimization.

We then evaluated the proposed optimization approaches in more detail with various scenarios having a higher traffic demand at DTW, by analyzing several airport performance metrics. Both optimization approaches provided significant taxi-out time savings with no impacts on departure runway performance and on arrivals. In most traffic scenarios, the integrated approach performed a little better in terms of taxi-out time and takeoff delay, but its computational tractability was worse than with the three-step approach. When we used the three-step approach, minimizing

the makespan would have been a better objective in the runway scheduling phase because it required shorter computation time and less controller workload, while the taxi times and takeoff delays obtained after taxiway scheduling were the same as the minimum runway delay solution. We also showed that the traffic characteristics in the flight schedule, such as fleet mix ratio and demand fluctuation, could affect the airport performance metrics of departures, including gate-holding time, takeoff delay, and position changes in takeoff sequencing. However, the optimized taxi times obtained by both optimization approaches were independent of the traffic properties and close to the unimpeded taxi times.

Next, we compared two types of departure control methods for efficient surface traffic management, which were aggregate queue-based control (N-Control) and individual aircraft trajectory-based control (RTA control). If implementable, the RTA control based on the optimal taxiway schedules from the surface traffic optimization would provide us with the most significant taxi time savings without losing any runway performance. Since we are not able to realize the RTA control in the current operational environment, we modeled an interim control method using the optimized pushback times only among the RTAs as control inputs in the SIMMOD simulations, and this pushback time control method (PbT control) was still able to reduce the taxi times considerably. The SIMMOD simulations for the N-Control method demonstrated that as we lowered the queue capacity control parameter, we could obtain the better taxiway performance, while maintaining the same level of runway throughput. However, the taxi time savings from N-Control could not exceed the benefits from the trajectory-based control because the N-Control method was activated only when the surface traffic was congested. We also found that the departure control using the gate-holding strategy could result in gate conflicts with arrivals. We examined possible solutions to avoid the gate conflicts through fast-time simulations. The evaluation results suggested that the earlier pushback of departures could be a better option to minimize side effects than the waiting of arrivals.

In the last part of the thesis, we investigated how the uncertainty factors in airport operations could have impacts on airport performance. For this evaluation, we de-

defined the main uncertainties that might affect the airport performance and developed stochastic simulation models using SIMMOD for these uncertainty sources. Monte Carlo simulations for each uncertainty factor were then conducted with various flight schedules at DTW. Simulation results showed that the ground delays saturated or decreased as the uncertainty in pushback times grew because the unintended pushback delay due to the uncertainty might act as the gate-holding strategy. Uncertain runway exit times for arrivals did not significantly impact airport performance, apart from gate arrival times. It was also found that taxi speed variations among aircraft significantly increased ground delay. Uncertainty in inter-departure separation times increased wait times in the departure queues, while reducing runway throughput. These uncertainty studies also compared the simulations of originally scheduled and optimized pushback times and showed that the surface traffic optimization based on a deterministic model could provide benefits even in the presence of certain types of uncertainties.

6.2 Future research directions

The research work in this thesis can be expanded in the following directions in the future.

6.2.1 Extensions of airport surface traffic optimization

The objectives used in the airport surface traffic optimization approaches proposed in this thesis can be varied by considering various stakeholders related to airport operations. Instead of minimizing taxi times and runway delays or maximizing runway throughput, we may want to minimize operational costs for airlines or environmental impacts on the surface such as fuel consumption and gas emissions. These variants can be calculated easily by changing the delay cost coefficients for individual flights in the objective function of the given optimization model, as long as appropriate data are furnished. We can also refine the operational parameters used in the optimization to enable more accurate control of aircraft on the surface. Although the current

optimization models are based on reasonable assumptions, the values for the nominal taxi speed, the minimum separation distance on the taxiway, and the maximum holding time could be further subdivided, depending on the aircraft weight classes or models. For the runway scheduling of departures, we may consider more separation constraints like miles-in-trail (MIT) and departure fixes after takeoffs as well.

In this thesis, we assumed that the taxi routes of flights were given. However, flights may have alternative routes to reach the assigned runway or gate. By introducing decision variables for selecting the optimal taxi route among available routes, we can extend the existing MILP models to deal with the multiple route taxi scheduling problem, as proposed in [94]. In the three-step approach, for instance, the MILP model in Step 3 may be modified to choose a better taxi route for each aircraft while meeting the optimized takeoff times from the runway scheduling algorithm in Step 2. This approach can also be extended to assign departures to the better runway, when the airport has multiple departure runways. In addition, for efficient terminal area operations, the airport surface traffic optimization should be integrated properly with the traffic flow management problem in the airspace.

We assumed that the airport runway configuration did not change during the evaluation because the current configuration is the most frequently used one and the configuration changes do not often happen at DTW. In general, however, the runway configuration can change due to several factors affecting airport operations such as weather conditions, wind direction, airspace availability, noise mitigation procedures, and traffic demand [37]. It is therefore necessary to consider how the runway and taxiway scheduling algorithms can work in the configuration change situation. During airport configuration changes, we may use the runways used in the previous runway configuration as taxiways for rerouting and optimize the runway and taxiway schedules again attuned to the new configuration, as suggested in [37]. In this approach, the departure flights which have left the gates, but have not yet taken off will be rescheduled along the temporary taxi routes from their current locations to the new departure runways.

Improving the computational performance of the optimization can be a significant

research direction for future work. We can introduce mathematical approaches like Bender's decomposition to the current optimization model, but we may be able to reduce the solution time more easily by appropriate problem modeling techniques. It would be helpful to reduce the number of control points on the surface by using a simplified node-link network model for the airport layout or excluding the ramp area from the control domain. Also, making the time discretization rougher could improve the computational performance significantly at a relatively small cost to the optimization benefit, as shown in the related tradeoff study in [85]. These efforts would make the proposed optimization approaches more amenable to practical use as a decision support tool for real airport operations.

6.2.2 Adaptations to other airports

In this thesis, we focused on the surface traffic optimization at Detroit airport, but we can apply the same approaches to other busy airports. For these applications, we have to create a node-link network model to represent the target airport layout based on the geographical information first. We also need to carefully consider the surface infrastructure characteristics and operational rules of the airport, such as runway configuration, taxiway layouts, holding area locations, standard taxi routes, available taxi speed range, separation requirement rules, and gate usage limitations. If there are lots of interactions between departures and arrivals on such taxiway areas like intersections and runway crossings, we will be able to obtain significant taxi time savings through optimization. On the other hand, if the airport has a simple layout with a single runway for departures and few interactions with arrivals, it will likely derive benefits from departure runway scheduling alone. For the large airports having multiple runways and a complicated airport layout, the proposed surface traffic optimization approaches may be more effective because there are more opportunities to optimize the flight schedules by sequencing at control points.

The traffic characteristics of the target airport such as traffic demand level and fleet mix ratio can also affect the benefits of the scheduling algorithms, as shown in Chapter 3. We expect greater benefits from optimization during peak traffic times. If

the fleet mix ratio in the flight schedule becomes more heterogeneous among aircraft types, we can reduce takeoff delay and taxi times more by the surface traffic optimization. In addition to the traffic characteristics tested in this thesis, the optimization models can be evaluated with other traffic scenarios (e.g., impact of general aviation, addition of A380 aircraft) for further study in the future.

6.2.3 Applications of fast-time air traffic simulations

We showed that the fast-time simulations using SIMMOD could be used effectively for evaluating the new control approaches to improving surface traffic management and analyzing the statistical data from various traffic scenarios in the presence of uncertainties before implementing human-in-the-loop (HITL) simulations or field tests. If the simulations present sufficient evidence of reliability through validation with historical data, we can try simulation-based optimization to find the optimal pushback times of departures by repeating a number of simulations with updated inputs until the objective reaches an acceptable level [99].

Through the stochastic simulations using SIMMOD, we can also evaluate the other sources of uncertainty in airport and airspace operations (e.g., weather conditions, ceiling/visibility, airport configuration, departure/arrival fixes) in a similar manner. Since these uncertainties usually occur together in the real world, the combined impacts of different sources of uncertainty will need to be studied as well. As the main sources of uncertainty are different from one airport to another, we also need to implement the same uncertainty studies for the other airports so as to check the impact of the airport characteristics.

Bibliography

- [1] J. Abela, D. Abramson, M. Krishnamoorthy, A. De Silva, and G. Mills. Computing Optimal Schedules for Landing Aircraft. In *The 12th National ASOR Conference*, pages 71–90, Adelaide, Australia, 1993.
- [2] AirportTools. Basic SIMMOD Study Creation Tutorial, March 2005. <http://www.airporttools.com/>.
- [3] I. Anagnostakis and J.-P. Clarke. Runway Operations Planning: a Two-Stage Solution Methodology. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*. IEEE, 2003.
- [4] I. Anagnostakis, J.-P. Clarke, D. Böhme, and U. Völckers. Runway Operations Planning and Control: Sequencing and Scheduling. *Journal of Aircraft*, 38(6), 2001.
- [5] I. Anagnostakis, H. R. Idris, J.-P. Clarke, E. Feron, R. J. Hansman, A. R. Odoni, and W. D. Hall. A Conceptual Design of A Departure Planner Decision Aid. In *3rd USA/Europe Air Traffic Management R&D Seminar*, Napoli, Italy, June 2000.
- [6] R. Anderson and D. Milutinović. An Approach to Optimization of Airport Taxiway Scheduling and Traversal under Uncertainty. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 227(2):273–284, 2013.
- [7] ATAC Corporation. SIMMOD Reference Manual, 1995. <http://www.atac.com/>.
- [8] J. A. D. Atkin, E. K. Burke, and S. Ravizza. The Airport Ground Movement Problem: Past and Current Research and Future Directions. In *Proceedings of the 4th International Conference on Research in Air Transportation (ICRAT)*, Budapest, Hungary, June 2010.
- [9] Jason A.D. Atkin. *Airport Airside Optimisation Problems In Automated Scheduling and Planning Studies in Computational Intelligence*, volume 505, pages 1–37. Springer Berlin Heidelberg, 2013.

- [10] S. Atkins, C. Brinton, and Y. Jung. Implications of Variability in Airport Surface Operations on 4-D Trajectory Planning. In *AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Anchorage, AK, September 2008.
- [11] S. Atkins, B. Capozzi, A. Churchill, A. Fernandes, and C. Provan. Investigation of NASA's Spot and Runway Departure Advisor Concept at PHL, CLT, and LAX Airports. In *10th USA/Europe Air Traffic Management R&D Seminar*, Chicago, IL, June 2013.
- [12] H. Baik and A. A. Trani. Framework of a Time-Based Simulation Model for the Analysis of Airfield Operations. *Journal of Transportation Engineering*, 134(10):397-413, 2008.
- [13] H. Balakrishnan and B. Chandran. Scheduling Aircraft Landings under Constrained Position Shifting. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, CO, 2006.
- [14] H. Balakrishnan and B. Chandran. Efficient and Equitable Departure Scheduling in Real-time: New Approaches to Old Problems. In *7th USA/Europe Air Traffic Management R&D Seminar*, Barcelona, Spain, July 2007.
- [15] H. Balakrishnan and B. Chandran. Algorithms for Scheduling Runway Operations under Constrained Position Shifting. *Operations Research*, 58(6):1650-1665, 2010.
- [16] H. Balakrishnan and Y. Jung. A Framework for Coordinated Surface Operations Planning at Dallas-Fort Worth International Airport. In *AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, August 2007.
- [17] C. Barnhart, D. Fearing, A. Odoni, and V. Vaze. Demand and Capacity Management in Air Transportation. *EURO Journal of Transportation and Logistics*, 1(1-2):135-155, 2012.
- [18] A. M. Bayen, C. J. Tomlin, Y. Ye, and J. Zhang. An Approximation Algorithm for Scheduling Aircraft with Holding Time. In *IEEE Conference on Decision and Control (CDC)*, 2004.
- [19] J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson. Scheduling Aircraft Landings - The Static Case. *Transportation Science*, 34(2):180-197, 2000.
- [20] J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson. Displacement Problem and Dynamically Scheduling Aircraft Landings. *Journal of the Operational Research Society*, 55:54-64, 2004.
- [21] Peter Belobaba, Amadeo Odoni, and Cynthia Barnhart. *The Global Airline Industry*. Wiley, 2009.

- [22] D. Bertsimas and S. S. Patterson. The Air Traffic Flow Management Problem with Enroute Capacities. *Operations Research*, 46(3):406–422, 1998.
- [23] L. Bianco, S. Ricciardelli, G. Rinaldi, and A. Sassano. Scheduling Tasks with Sequence-dependent Processing Times. *Naval Research Logistics*, 35(2):177–184, 1988.
- [24] L. Bianco, G. Rinaldi, and A. Sassano. A Combinatorial Optimization Approach to Aircraft Sequencing Problem. In A. R. Odoni, L. Bianco, and G. Szego, editors, *Flow Control of Congested Networks*, volume 38 of *NATO ASI Series, Series F: Computer Systems and Systems Science*, pages 323–339, Springer-Verlag, Berlin, 1987.
- [25] D. Böhme. Tactical Departure Management with the Eurocontrol/ DLR DMAN. In *6th USA/Europe Air Traffic Management R&D Seminar*, Baltimore, MD, 2005.
- [26] C. Brinton and S. Atkins. A Probabilistic Modeling Foundation for Airport Surface Decision Support Tools. In *Integrated Communication, Navigation and Surveillance Conference*, Arlington, VA, May 2009.
- [27] C. Brinton, C. Provan, S. Lent, T. Prevost, and S. Passmore. Collaborative Departure Queue Management: An Example of Collaborative Decision Making in the United States. In *9th USA/Europe Air Traffic Management R&D Seminar*, Berlin, Germany, June 2011.
- [28] C. R. Brinton, S. C. Atkins, and A. Hall. Analysis of Taxi Conformance Monitoring Algorithms and Performance. In *Proceedings of the Integrated Communication, Navigation and Surveillance Conference*, Dulles, Virginia, 2007.
- [29] C. R. Brinton, L. S. Cook, and S. C. Atkins. Collaborative Airport Surface Metering for Efficiency and Environmental Benefits. In *Proceedings of the Integrated Communication, Navigation and Surveillance Conference*, Dulles, Virginia, 2007.
- [30] P. Burgain. *On the Control of Airport Departure Processes*. PhD thesis, Georgia Institute of Technology, 2010.
- [31] P. Burgain, E. Feron, and J.-P. Clarke. Collaborative Virtual Queue: Benefit Analysis of a Collaborative Decision Making Concept Applied to Congested Airport Departure Operations. *Air Traffic Control Quarterly Journal*, 17(2):195–222, 2009.
- [32] F. R. Carr. *Robust Decision-Support Tools for Airport Surface Traffic*. PhD thesis, Massachusetts Institute of Technology, 2004.
- [33] B. Chandran and H. Balakrishnan. A Dynamic Programming Algorithm for Robust Runway Scheduling. In *American Control Conference*, New York, NY, July 2007.

- [34] V. H. L. Cheng. Airport Surface Operation Collaborative Automation Concept. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Austin, TX, August 2003.
- [35] V. H. L. Cheng, G. D. Sweriduk, J. Yeh, A. D. Andre, and D. C. Foyle. Flight-Deck Automation for Trajectory-Based Surface Operations. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Honolulu, HI, August 2008.
- [36] G. Clare, A. Richards, and S. Sharma. Receding Horizon, Iterative Optimization of Taxiway Routing and Runway Scheduling. In *AIAA Guidance, Navigation, and Control Conference*, Chicago, IL, August 2009.
- [37] J.-P. Clarke, L. Li, H. Balakrishnan, E. Feron, K. Griffin, B. Kim, S. Kim, H. Lee, I. J. Robeson, G. Solveling, P. Yu, and J. Brook. Surface Traffic Optimization in the Presence of Uncertainties. Final report, GaTech Surface Team, NASA Project NNX07AU34A, Atlanta, GA, December 2010.
- [38] W. W. Cooper, R. H. Cormier, J. G. Foster, M. J. Mills, and S. C. Mohleji. Use of the Departure Enhanced Planning and Runway/Taxiway Assignment System (DEPARTS) for optimal departure scheduling at busy airports. In *Digital Avionics Systems Conference*, 2002.
- [39] Richard de Neufville and Amadeo Odoni. *Airport Systems: Planning, Design and Management*. McGraw-Hill, 2003.
- [40] R. G. Dear. *The Dynamic Scheduling of Aircraft in the Near Terminal Area*. PhD thesis, Massachusetts Institute of Technology, 1976. Flight Transportation Laboratory Report R76-9.
- [41] R. G. Dear and Y. S. Sherif. An Algorithm for Computer Assisted Sequencing and Scheduling of Terminal Area Operations. *Transportation Research, Part A, Policy and Practice*, 25:129-139, 1991.
- [42] R. Deau, J. B. Gotteland, and N. Durand. Runways Sequences and Ground Traffic Optimisation. In *Proceedings of the 3rd International Conference on Research in Air Transportation*, Fairfax, VA, June 2008.
- [43] R. Deau, J. B. Gotteland, and N. Durand. Airport Surface Management and Runways Scheduling. In *Proceedings of the 8th USA/Europe Air Traffic Management R&D Seminar*, Napa, CA, 2009.
- [44] B. Delcaire and E. Feron. Development of an On-site Ground Operations Model for Logan International Airport. final report, FAA Air Transportation Center of Excellence in Operations Research, MIT, Cambridge, MA, December 1997. <http://www.isr.umd.edu/NEXTOR/pubs/RR-97-9.pdf>.
- [45] A. T. Ernst, M. Krishnamoorthy, and R. H. Storer. Heuristic and Exact Algorithms for Scheduling Aircraft Landings. *Networks*, 34:229-241, 1999.

- [46] F. Neuman and H. Erzberger. Analysis of Sequencing and Scheduling Methods for Arrival Traffic. NASA Technical Memorandum 102795, April, 1990.
- [47] Federal Aviation Administration (FAA). Air Traffic Control: Order JO 7110.65U. Includes Change 2, effective March 07, 2013.
- [48] Federal Aviation Administration (FAA). Airport Capacity Benchmark Report 2004, 2004.
- [49] E. R. Feron, R. J. Hansman, A. R. Odoni, R. B. Cots, B. Delcaire, X. Feng, W. D. Hall, H. R. Idris, A. Muharremoglu, and N. Pujet. The Departure Planner: A Conceptual Discussion. In *Technical report*. Massachusetts Institute of Technology, December 1997.
- [50] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. The Scientific Press (now part of Wadsworth Publishing), 1993.
- [51] M. J. Frankovich. *Air Traffic Flow Management at Airports: a Unified Optimization Approach*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, August 2012.
- [52] W. Gao and S. Jiang. Simulation Study on Closely Spaced Parallel Runway Analysis Using SIMMOD Plus. In *International Conference on Intelligent Computation Technology and Automation*, pages 344–347, Changsha, China, May 2010.
- [53] W. Gao and S. Xu. Multi-Airport System Simulation and Optimization Research. In *International Conference on Electronic Computer Technology*, pages 698–701, Macau, China, February 2009.
- [54] W. Gao, X. Xu, L. Diao, and H. Ding. SIMMOD Based Simulation Optimization of Flight Delay Cost for Multi-airport System. In *Intelligent Computation Technology and Automation (ICICTA)*, pages 698–702, Hunan, China, October 2008.
- [55] C. Gong. Kinematic Airport Surface Trajectory Model Development. In *AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Hilton Head, SC, September 2009.
- [56] J. B. Gotteland and N. Durand. Genetic Algorithms Applied to Airport Ground Traffic Optimization. In *Proceedings of the Congress on Evolutionary Computation*, volume 1, Canberra, Australia, 2003.
- [57] J. B. Gotteland, N. Durand, and J. M. Alliot. Handling CFMU Slots in Busy Airports. In *Proceedings of the 5th USA/Europe Air Traffic Management R&D Seminar*, Budapest, Hungary, 2003.

- [58] J. B. Gotteland, N. Durand, J. M. Alliot, and E. Page. Aircraft Ground Traffic Optimization. In *Proceedings of the 4th USA/Europe Air Traffic Management R&D Seminar*, Santa Fe, NM, 2001.
- [59] K. J. Griffin, P. Yu, and D. B. Rappaport. Evaluating Surface Optimization Techniques Using a Fast-time Airport Surface Simulation. In *AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Fort Worth, TX, September 2010.
- [60] G. Gupta, W. Malik, and Y. Jung. A Mixed Integer Linear Program for Airport Departure Scheduling. In *AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Hilton Head, SC, September 2009.
- [61] G. Gupta, W. Malik, and Y. Jung. Incorporating Active Runway Crossings in Airport Departure Scheduling. In *AIAA Guidance, Navigation and Control Conference*, Toronto, Ontario, Canada, August 2010.
- [62] G. Gupta, W. Malik, and Y. C. Jung. Effect of Uncertainty on Deterministic Runway Scheduling. In *AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Virginia Beach, VA, September 2011.
- [63] G. Gupta, W. Malik, and Y. C. Jung. An Integrated Collaborative Decision Making and Tactical Advisory Concept for Airport Surface Operations Management. In *AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Indianapolis, IN, September 2012.
- [64] G. Gupta, W. Malik, L. Tobias, Y. Jung, T. Hoang, and M. Hayashi. Performance Evaluation of Individual Aircraft Based Advisory Concept for Surface Management. In *10th USA/Europe Air Traffic Management R&D Seminar*, Chicago, IL, June 2013.
- [65] M. Hayashi, T. Hoang, Y. Jung, G. Gupta, W. Malik, and V. Dulchinos. Usability Evaluation of the Spot and Runway Departure Advisor (SARDA) Concept in a Dallas/Fort Worth Airport Tower Simulation. In *10th USA/Europe Air Traffic Management R&D Seminar*, Chicago, IL, June 2013.
- [66] R. Horonjeff and F. X. McKelvey. *Planning and Design of Airports, 4th Edition*. McGraw-Hill, 1993.
- [67] D. W. Hutchison and S. D. Hill. Simulation Optimization of Airline Delay with Constraints. In *Proceedings of the Winter Simulation Conference*, volume 2, pages 1017–1022, Arlington, VA, December 2001.
- [68] H. Idris, J.-P. Clarke, R. Bhuva, and L. Kang. Queuing Model for Taxi-Out Time Estimation. In *ATC Quarterly*, September 2001.
- [69] H. R. Idris, B. Delcaire, I. Anagnostakis, W. D. Hall, N. Pujet, E. Feron, R. J. Hansman, J.-P. Clarke, and A. R. Odoni. Identification of Flow Constraint and

- Control Points in Departure Operations at Airport Systems. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Boston, MA, August 1998.
- [70] ILOG, Inc. ILOG CPLEX 11.0.1, 2008. <http://www-01.ibm.com/software/websphere/ilog/>.
- [71] A. Jacquillat. A Queuing Model of Airport Congestion and Policy Implications at JFK and EWR. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, June 2012.
- [72] Joint Economic Committee. Your Flight Has Been Delayed Again, May, 2008.
- [73] R. Jordan, M. Ishutkina, and T. Reynolds. A Statistical Learning Approach to the Modeling of Aircraft Taxi Time. In *Digital Avionics Systems Conference*, Salt Lake City, UT, October 2010.
- [74] Y. Jung, T. Hoang, J. Montoya, G. Gupta, W. Malik, and L. Tobias. A Concept and Implementation of Optimized Operations of Airport Surface Traffic. In *AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Fort Worth, TX, September 2010.
- [75] Y. Jung, T. Hoang, J. Montoya, G. Gupta, W. Malik, L. Tobias, and H. Wang. Performance Evaluation of a Surface Traffic Management Tool for Dallas/Fort Worth International Airport. In *9th USA/Europe Air Traffic Management R&D Seminar*, Berlin, Germany, June 2011.
- [76] G. Keith, A. Richards, and S. Sharma. Optimization of Taxiway Routing and Runway Scheduling. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Honolulu, HI, August 2008.
- [77] H. Khadilkar and H. Balakrishnan. Estimation of Aircraft Fuel Burn Using Flight Data Recorder Archives. In *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, August 2011.
- [78] S. H. Kim and E. Feron. Robust Gate Assignment. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR, August 2011.
- [79] S. H. Kim and E. Feron. Impact of Gate Assignment on Gate-holding Departure Control Strategies. In *Digital Avionics Systems Conference*, Williamsburg, VA, October 2012.
- [80] N. L. Kleinman, S. D. Hill, and V. A. Ilenda. SPSA/SIMMOD Optimization of Air Traffic Delay Cost. In *Proceedings of the American Control Conference*, pages 1121–1125, Albuquerque, NM, June 1997.
- [81] N. L. Kleinman, S. D. Hill, and V. A. Ilenda. Simulation Optimization of Air Traffic Delay Cost. In *Proceedings of the Winter Simulation Conference*, pages 1177–1181, Washington, DC, December 1998.

- [82] H. Lee. Tradeoff Evaluation of Scheduling Algorithms for Terminal Area Air Traffic Control. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, June 2008.
- [83] H. Lee and H. Balakrishnan. A Study of Tradeoffs in Scheduling Terminal-Area Operations. In *Proceedings of the IEEE*, volume 96, No. 12, December 2008.
- [84] H. Lee and H. Balakrishnan. Fuel Cost, Delay and Throughput Tradeoffs in Runway Scheduling. In *Proceedings of the American Control Conference*, Seattle, WA, June 2008.
- [85] H. Lee and H. Balakrishnan. Optimization of Airport Taxiway Operations at Detroit Metropolitan Airport (DTW). In *AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Fort Worth, TX, September 2010.
- [86] H. Lee and H. Balakrishnan. A Comparison of Two Optimization Approaches for Airport Taxiway and Runway Scheduling. In *Digital Avionics Systems Conference*, Williamsburg, VA, October 2012.
- [87] H. Lee, I. Simaiakis, and H. Balakrishnan. A Comparison of Aircraft Trajectory-based and Aggregate Queue-based Control of Airport Taxi Processes. In *Digital Avionics Systems Conference*, Salt Lake City, UT, October 2010.
- [88] A. Lim and F. Wang. Robust Airport Gate Assignment. In *17th IEEE International Conference on Tools with Artificial Intelligence*, pages 74–81, Hong Kong, November 2005.
- [89] W. Malik, G. Gupta, and Y. Jung. Managing Departure Aircraft Release for Efficient Airport Surface Operations. In *AIAA Guidance, Navigation and Control Conference*, Toronto, Ontario, Canada, August 2010.
- [90] W. Malik, G. Gupta, and Y. C. Jung. Spot Release Planner: Efficient Solution for Detailed Airport Surface Traffic Optimization. In *AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Indianapolis, IN, September 2012.
- [91] N. Martel. SIMMOD-Based Simulation Montreal Dorval (YUL) Case Study. TRB Workshop on Airport-Airspace Simulations for Capacity Evaluation, January 2000. <http://onlinepubs.trb.org/onlinepubs/circulars/ec035/09martel.pdf>.
- [92] V. Mehta, M. E. Miller, T. Reynolds, M. Ishutkina, R. Jordan, R. Seater, and W. Moser. Decision Support Tools for the Tower Flight Data Manager System. In *Integrated Communications, Navigation and Surveillance Conference (ICNS)*, Herndon, VA, May 2011.
- [93] J. Montoya, Z. Wood, and S. Rathinam. Runway Scheduling Using Generalized Dynamic Programming. In *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, 2011.

- [94] J. Montoya, Z. Wood, S. Rathinam, and W. Malik. A Mixed Integer Linear Program for Solving a Multiple Route Taxi Scheduling Problem. In *AIAA Guidance, Navigation, and Control Conference*, Toronto, Ontario, Canada, August 2010.
- [95] A. Nakahara, T. Reynolds, T. White, C. Maccarone, and R. Dunskey. Analysis of a Surface Congestion Management Technique at New York JFK Airport. In *AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Virginia Beach, VA, September 2011.
- [96] F. Neuman and H. Erzberger. Analysis of Delay Reducing and Fuel Saving Sequencing and Spacing Algorithms for Arrival Spacing. NASA Technical Memorandum 103880, October, 1991.
- [97] A. R. Odoni, J. Bowman, D. Delahaye, J. J. Deyst, E. Feron, R. J. Hansman, K. Khan, J. K. Kuchar, N. Pujet, and R. W. Simpson. Existing and Required Modeling Capabilities for Evaluating ATM System and Concepts. Technical report, International Center for Air Transportation, Massachusetts Institute of Technology, Cambridge, MA, March 1997. <http://catsr.ite.gmu.edu/ATSE/OdoniModelingResearchReview.pdf>.
- [98] A. R. Odoni, J.-M. Rousseau, and N. H. M. Wilson. Chapter 5. Models in Urban and Air Transportation. In S. M. Pollock, M. H. Rothkopf, and A. Barnett, editors, *Operations Research and the Public Sector: Handbooks in Operations Research and Management Science*, volume 6. Elsevier Science B. V., 1994.
- [99] C. Osorio and M. Bierlaire. A Simulation-based Optimization Framework for Urban Traffic Control. *Proceedings of the Annual Transportation Research Forum (TRF)*, 2010.
- [100] B. Pesic, N. Durand, and J. M. Alliot. Aircraft Ground Traffic Optimisation Using a Genetic Algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference*, San Francisco, CA, 2001.
- [101] L. J. Prinzel, D. R. Jones, K. J. Shelton, J. J. Arthur, R. E. Bailey, A. S. Allamandola, D. C. Foyle, and B. L. Hooey. Flight Deck Display Technologies for 4DT and Surface Equivalent Visual Operations. *Proceedings of the 15th International Symposium on Aviation Psychology*, 2009.
- [102] H. N. Psaraftis. A Dynamic Programming Approach for Sequencing Groups of Identical Jobs. *Operations Research*, 28:1347-1359, 1980.
- [103] N. Pujet, B. Delcaire, and E. Feron. Input-Output Modeling and Control of the Departure Process of Congested Airports. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pages 1835-1852, Portland, OR, 1999.

- [104] S. Rathinam, J. Montoya, and Y. Jung. An Optimization Model for Reducing Aircraft Taxi Times at the Dallas Fort Worth International Airport. In *26th International Congress of the Aeronautical Sciences*, Anchorage, AK, September 2008.
- [105] P. C. Roling. Airport Surface Traffic Planning Optimization: a Case Study of Amsterdam Airport Schiphol. In *AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Hilton Head, SC, September 2009. AIAA 2009-7079.
- [106] P. C. Roling. TPMagic, a Universal Airport Surface Traffic Planning Analysis and Optimization Tool. In *AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Virginia Beach, VA, September 2011. AIAA 2011-7006.
- [107] P. C. Roling and H. G. Visser. Optimal Airport Surface Traffic Planning Using Mixed-Integer Linear Programming. *International Journal of Aerospace Engineering*, 2008. Article ID 732828.
- [108] K. Roy, A. M. Bayen, and C. J. Tomlin. Polynomial Time Algorithms for Scheduling of Arrival Aircraft. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, CA, August 2005.
- [109] M. S. Ryerson, M. Hansen, and J. Bonn. Fuel Consumption and Operational Performance. In *9th USA/Europe Air Traffic Management R&D Seminar*, Berlin, Germany, June 2011.
- [110] M. Schaper, G. Tsoukala, R. Stavрати, and N. Papadopoulos. Departure Flow Control through Takeoff Sequence Optimisation: Setup and Results of Trials at Athens Airport. In *Digital Avionics Systems Conference*, Seattle, WA, October 2011.
- [111] R. Shumsky. *Dynamic Statistical Models for the Prediction of Aircraft Take-off Times*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, September 1995.
- [112] I. Simaiakis. Modeling and Control of Airport Departure Processes for Emissions Reduction. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, September 2009.
- [113] I. Simaiakis. *Analysis, Modeling and Control of the Airport Departure Process*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, August 2012.
- [114] I. Simaiakis and H. Balakrishnan. Queuing Models of Airport Departure Processes for Emissions Reduction. In *AIAA Guidance, Navigation, and Control Conference*, Chicago, IL, August 2009.

- [115] I. Simaiakis and H. Balakrishnan. Impact of Congestion on Taxi Times, Fuel Burn and Emissions at Major Airports. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 2184, pages 22–30, December 2010.
- [116] I. Simaiakis, H. Khadilkar, H. Balakrishnan, T. Reynolds, R. J. Hansman, B. Reilly, and S. Urlass. Demonstration of Reduced Airport Congestion Through Pushback Rate Control. In *9th USA/Europe Air Traffic Management R&D Seminar*, Berlin, Germany, June 2011.
- [117] J. W. Smeltink, M. J. Soomer, P. R. de Waal, and R. D. van der Mei. An Optimisation Model for Airport Taxi Scheduling. In *Thirtieth Conference on the Mathematics of Operations Research*, Lunteren, The Netherlands, January 2005.
- [118] G. Solveling. *Stochastic Programming Methods for Scheduling of Airport Runway Operations Under Uncertainty*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, August 2012.
- [119] G. Solveling, S. Solak, J.-P. Clarke, and E. Johnson. Runway Operations Optimization in the Presence of Uncertainties. In *AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, Fort Worth, TX, September 2010.
- [120] A. A. Trani. Runway Occupancy Time Estimation and SIMMOD. North American SIMMOD User’s Group Meeting, Chicago, IL, September 2000. <http://www.nasug.com/200009/rot.pdf>.
- [121] A. A. Trani, H. Baik, J. Martinez, and V. Kamat. A New Paradigm to Model Aircraft Operations at Airports: The Virginia Tech Airport SIMulation Model (VTASIM). In *Nextor Fourth Annual Research Symposium*, FAA Headquarters Auditorium, November 2000.
- [122] D. A. Trivizas. Optimal Scheduling with Maximum Position Shift (MPS) Constraints: A Runway Scheduling Application. *Journal of Navigation*, 51(2):250–266, May 1998.
- [123] C. S. Venkatakrishnan. *Analysis and Optimization of Terminal Area Air Traffic Control Operations*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, February 1991.
- [124] C. S. Venkatakrishnan, A. Barnett, and A. R. Odoni. Landings at Logan Airport: Describing and Increasing Airport Capacity. *Transportation Science*, 27(3):211–227, 1993.
- [125] H. G. Visser and P. C. Roling. Optimal Airport Surface Traffic Planning Using Mixed Integer Linear Programming. In *AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Denver, CO, 2003.

- [126] L. Xu, F. Wang, and Z. Xu. A Robust Approach for the Airport Gate Assignment. In *Proceedings of International Forum on Shipping, Ports and Airports*, pages 15–24, Chengdu, China, October 2010.