INVITED PAPER

A Study of Tradeoffs in Scheduling Terminal-Area Operations

Scheduling the arrival of air traffic at airports involves tradeoffs between traffic throughput, arrival delays, and the costs of operations and fuel.

By Hanbong Lee, *Student Member IEEE*, and Hamsa Balakrishnan, *Member IEEE*

ABSTRACT | The terminal area surrounding an airport is an important component of the air transportation system, and efficient terminal-area schedules are essential for accommodating the projected increase in air traffic demand. Aircraft arrival schedules are subject to a variety of operational constraints, such as minimum separation for safety, required arrival time-windows, limited deviation from a first-come first-served sequence, and precedence constraints. There is also a range of objectives associated with multiple stakeholders that could be optimized in these schedules; the associated tradeoffs are evaluated in this paper. A dynamic programming algorithm for determining the minimum cost arrival schedule, given aircraft-dependent delay costs, is presented. The proposed approach makes it possible to determine various tradeoffs in terminal-area operations. A comparison of maximum throughput and minimum average delay schedules shows that the benefit from maximizing throughput could be at the expense of an increase in average delay, and that minimizing delay is the more advantageous of the two objectives in most cases. A comprehensive analysis of the tradeoffs between throughput and fuel costs and throughput and operating costs is conducted, accounting for both the cost of delay (as reported by the airlines) and the cost of speeding up when possible (from models of aircraft performance).

KEYWORDS | Air traffic management; air transportation systems; terminal-area scheduling and optimization; tradeoff evaluation

I. INTRODUCTION

Air traffic congestion is widely considered one of the principal constraints to the future growth of the global air transportation industry [1]. The Department of Transportation estimates that commercial aviation delays cost U.S. airlines more than $3 billion per year in direct operating costs alone [2]. With air transportation having become the backbone of global commerce and transporting 36% of all international freight, the indirect costs to passengers and businesses are much higher.[1] The demand for air traffic is expected to increase to between two and three times current values by the year 2025 [3]. As a result, congestion delays will increase unless new air traffic management (ATM) solutions are developed and implemented. This realization has motivated research efforts in both the United States [Next Generation Air Transportation System (NGATS or NextGen)] and Europe [Single European Sky ATM Research (SESAR)] [4].[2]

An important step toward successfully meeting the increased demand for air traffic services is increasing the efficiency of arrival and departure operations. The terminal area forms the critical interface between the airspace and the airport (surface). Safe and efficient arrival and departure operations at airports are therefore important for the smooth functioning of the air transportation system. While efficiency is important, airport runway schedules are limited by the different operational constraints that are imposed by the system, such as separation

Manuscript received October 29, 2007; revised April 19, 2008. Current version published January 16, 2009. This work was supported by the NGATS-ATM Airspace and NGATS-ATM Airportal Programs, NASA, under Contracts NNA06CN24A and R8771-G1, and by NSF under Contract ECCS-0745237.

H. Lee is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: hanbong@mit.edu).

H. Balakrishnan is with the Department of Aeronautics and Astronautics and Engineering Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: hamsa@mit.edu).

Digital Object Identifier: 10.1109/JPROC.2008.2006145

[1] http://www.iata.org/index.htm.

[2] http://www.eurocontrol.int/sesar/.

0018-9219/$25.00 ©2008 IEEE

requirements for safety, air traffic controller flexibility, airline equity concerns, and the performance envelopes of the aircraft. In addition, there are several possible objectives that may need to be optimized while scheduling runways: from the perspective of air traffic control, throughput and average delay are important metrics, while from the airline perspective, the operating costs, especially fuel costs, are important. It is often necessary to make tradeoffs between these objectives while scheduling aircraft operations.

The terminal area is a dynamic and uncertain environment, with constant updates to aircraft states being obtained from surveillance systems and airline reports [5]. The dynamic nature of the terminal area necessitates the development of scheduling algorithms that are computationally efficient, and therefore amenable to replanning when new events occur or new data updates are obtained. The challenge of runway or terminal-area scheduling lies in simultaneously achieving safety, efficiency, and equity, which are often competing objectives [6]–[8], and doing so in a reasonable amount of time. While there is broad consensus on what constitutes safety (wake-vortex avoidance, downstream metering constraints), efficiency (high throughput, low average delay), and equity (limited deviation from the nominal order), as well as a large body of research, no solution approach has been able to adequately model and optimally solve the runway scheduling problem in a computationally tractable manner. One reason for this computational hurdle is that most runway scheduling models are, from a theoretical perspective, inherently hard to solve [9]. Consequently, most practical implementations resort to heuristic or approximate approaches that produce "good" solutions in a short time [6], [10], [11]. The difficulty in solving these scheduling models arises primarily because the solution space allows for the optimal sequence to deviate arbitrarily from the nominal sequence.

However, Dear [12] recognized that, in the short term, it was often unrealistic to allow large deviations from the nominal sequence for two reasons: i) the system may afford controllers limited flexibility in reordering aircraft and ii) large deviations from a nominal or "priority" schedule may be unacceptable to airlines from a fairness standpoint. This observation led to the constrained position shifting (CPS) framework for scheduling aircraft, which stipulated that an aircraft may be moved up to a specified maximum number of positions from its first-come first-served (FCFS) order. For example, if the maximum position shift allowed were two, an aircraft that is in the eighth position in the FCFS order can be placed at the sixth, seventh, eighth, ninth, or tenth position in the new order. Several researchers in both the United States and Europe have since used CPS to model fairness and worked toward developing fast solution techniques for scheduling within the CPS framework [1], [7], [13]. While some variants of CPS were shown to be solvable in polynomial time [14], [15], they were unable to handle all the operational constraints that arose in practice [16]. More importantly, these methods lack a unifying theory that allows their results to generalize to other interesting scheduling problems under CPS, even resulting in a conjecture that, in general, scheduling under CPS may have exponential complexity [7].

One possible approach to decreasing the average delay incurred by aircraft is to accelerate from their ideal speeds in order to arrive before their nominal or estimated time of arrival (ETA). This strategy, known as time advance (TA), is particularly beneficial during scenarios in which speeding up the first aircraft in a closely packed sequence decreases the delays incurred by the following aircraft [17], [18]. However, there are costs associated with TA, since the acceleration from the nominal speed results in greater fuel consumption by the aircraft. As a result, there is a point beyond which the cost of speeding up outweighs the benefit (in terms of delay reduction) of time advance. Taking this into account, Neuman and Erzberger used scheduling heuristics to show that CPS and TA methods could potentially increase fuel savings and decrease average delay [18].

In prior research, the scheduling problem of maximizing runway throughput under CPS was investigated [16]. A sequencing algorithm to take account of airline priorities was introduced [19]. A heuristic approach for minimizing the passenger-weighted sum of arrival times (with no time advance, arrival time-windows, or precedence constraints) also has been proposed [20]. However, to the best of our knowledge, there have been no comprehensive studies in recent years on the tradeoffs associated with terminal area schedules.

This research attempts to fill this gap by analyzing tradeoffs between multiple objectives of different stakeholders—for example, the tradeoff between throughput (for air traffic control) and operating costs (for airlines). For this evaluation, a dynamic programming based sequencing algorithm for optimizing a sum of general aircraft-dependent delay costs is developed by extending a framework previously used for determining the tradeoff between schedule robustness and throughput [21], [22]. Given arbitrary delay cost functions for each aircraft in the schedule, the proposed approach can determine the schedule that minimizes the total delay cost in computation time that scales linearly in the number of aircraft and as the square of the largest difference between the latest and earliest arrival time over all aircraft. The proposed algorithm considers both CPS and TA strategies simultaneously while optimizing the schedule and determines the optimal level of speedup for each aircraft in order to minimize the total fuel cost. In addition, Monte Carlo simulations are used to evaluate the tradeoff between throughput and average delay.

The remainder of this paper begins with the problem definition (Section II) followed by a description of the algorithmic framework (Section III). Section IV employs the framework to determine and analyze the tradeoffs between different objectives such as throughput, average delay, fuel costs, and operating costs. These tradeoffs are evaluated on real-world schedules based at Dallas Fort Worth (DFW) international airport.

II. PROBLEM DEFINITION

Given the cost of landing a particular aircraft at a particular time for a set of arriving aircraft, the problem is to determine the optimal arrival schedule that minimizes the sum of costs, referred to as the *total landing cost*. Runway schedules are also subject to several operational constraints, such as the limited flexibility afforded to air traffic controllers, available arrival time windows, minimum separation requirements, and precedence conditions.

A. Constraints

1) Limited Deviation From FCFS: The terminal area is an extremely dynamic environment, and resequencing aircraft increases the workload of controllers. Due to limited flexibility, it might not be possible for air traffic controllers to implement an efficient sequence that deviates significantly from the nominal or FCFS order. This is the basic motivation for CPS methods. CPS, first proposed by Dear [12], stipulates that an aircraft may be moved up to a specified maximum number of positions from its FCFS order. The maximum number of position shifts allowed is denoted by k ($k \leq 3$ for most runway systems), and the resulting environment is referred to as a k-CPS scenario. The restricted deviation from the FCFS order helps maintain equity among aircraft operators and also increases the predictability of landing times [16].

2) Arrival Time Windows: The airspace over the continental United States is divided into 20 geographical regions of air traffic control known as air route traffic control centers or, more simply, centers. Every airport is located within a center. Once an arriving aircraft is at the boundary of the center (about 45–60 min from the destination airport), tools such as the Trajectory Synthesizer (a decision-support system developed by NASA that predicts a complete time-based trajectory along the expected path [23]) may be used to determine the ETA at which the aircraft will land on an assigned runway, assuming it follows a nominal route and speed profile [18]. If the aircraft is speeded up, the actual time of arrival will be earlier than the estimated time of arrival. The earliest time of arrival is usually limited to one minute before the ETA because of the resultant fuel expenditure. The latest arrival time is determined either by fuel limitations or by the maximum delay that an aircraft can incur. The earliest and latest arrival times of aircraft i are denoted by $E(i)$ and $L(i)$, respectively.

3) Separation Requirements: The FAA regulates the minimum spacing between landing aircraft in order to avoid the danger of wake turbulence. The FAA classifies aircraft into three weight classes (heavy, large, and small) based on the maximum takeoff weight [24]. The FAA defines minimum separation distance requirements according to the weight classes of both the leading aircraft and the trailing aircraft during IFR approaches. These separation requirements in miles can be transformed to the minimum time separation required between landings, assuming a 5 nm final approach path and a nominal approach speed [1]. The matrix of minimum separation times (in seconds) is as follows:

Separation(in seconds)	**Trailing Aircraft**		
Leading Aircraft	Heavy	Large	Small
Heavy	96	157	196
Large	60	69	131
Small	60	69	82

The minimum required time between the leading aircraft i and the following aircraft j for arrivals is indicated by $\delta_{i,j}$.

4) Precedence Constraints: There could also be precedence constraints imposed on the landing sequence. These are constraints of the form aircraft i must land before aircraft j, and arise due to overtaking constraints, airline preferences from banking operations, or high-priority flights. Precedence relations are represented by a matrix $\{p_{ij}\}$ such that element $p_{ij} = 1$ if aircraft i must land before aircraft j, and $p_{ij} = 0$ otherwise.

B. Objective Function

There are several possible objective functions that may have to be optimized while determining arrival runway schedules. An important objective, maximizing runway throughput (or, alternatively, minimizing the completion time of a sequence of aircraft), was considered in prior work, and a dynamic programming based solution approach was proposed [16]. Minimizing the average delay or minimizing a weighted sum of delays, where the weights represent the relative priorities of flights (based on factors such as crew schedules, passenger and fleet connectivity, turnaround times, gate availability, on-time performance, fuel status, and runway assignments), are also desirable objectives [19]. There are inherent tradeoffs involved between these objectives, and the schedules that maximize throughput are not necessarily the same as those that minimize the average or the weighted sum of delays. For instance, when the cost per unit delay differs considerably between aircraft, the schedule with the minimum total landing cost may differ significantly from the schedule with the maximum throughput.

To encompass all the objective functions described above, a general landing cost function is introduced. Given a landing cost function $c_i(t_i)$, which corresponds to the cost of landing aircraft i at time t_i, the objective function is to minimize the sum of the landing costs of all aircraft in the schedule. Examples of the landing cost include the fuel cost (in dollars), the total fuel burn and the direct operating costs of the schedule. For instance, if the objective was to minimize the weighted sum of delays, where w_i was the weighting factor of aircraft i, then $c_i(t_i) = w_i t_i$. When all the weights are equal, minimizing the weighted sum is analogous to minimizing the total delay, or, equivalently, the average delay of the schedule.

C. Problem Statement

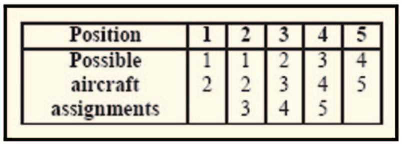
Integrating our objective and constraints, the problem of scheduling arrivals on a runway can be posed as follows.

Given n aircraft, earliest and latest arrival times $E(i)$ and $L(i)$ for the ith aircraft, separation matrix S, precedence matrix $\{p_{ij}\}$, costs $c_i(t_i)$ for aircraft i landing at time t_i, and the maximum number of position shifts k, we compute the optimal k-CPS sequence and corresponding landing times (t_i) to minimize the total landing cost, that is, the sum of the individual landing costs. Without loss of generality, aircraft can be labeled $(1,2,\ldots,n)$ according to their position in the FCFS sequence.

III. ALGORITHM

In prior work, it has been shown that every feasible k-CPS sequence can be represented as a path in a directed graph whose size is polynomially bounded in n and k [16]. This enables us to solve the problem at hand using dynamic programming, as is demonstrated in this section. We briefly describe the generation of this network and its properties.

Table 1 Possible Aircraft Assignments for $n = 5$, $k = 1$



Position	1	2	3	4	5
Possible aircraft assignments	1 2	1 2 3	2 3 4	3 4 5	4 5

A. The 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, $n = 5$ and $k = 1$, the nodes in stages $3, \ldots, 5$ represent all possible subsequences of length $2k + 1 = 3$ ending at that stage, while stage 1 contains a node for every possible sequence of length 1 ending (and starting) at position 1 and stage 2 contains a node for every possible sequence of length 2 ending at position 2. The network is generated using all possible aircraft assignments to each position in the sequence (Table 1).

For convenience, we refer to the last aircraft in a node's sequence as the *final aircraft* of that node. For each node in stage p, we draw directed arcs to all the nodes in stage p+1 that can follow it. Fig. 1 shows the network for $n = 5$ and $k = 1$. For example, node (2-1-3) in stage 3 is a successor of node (2-1) in the previous stage (stage 2) and can precede 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 sequence (2-1-3-4-5).

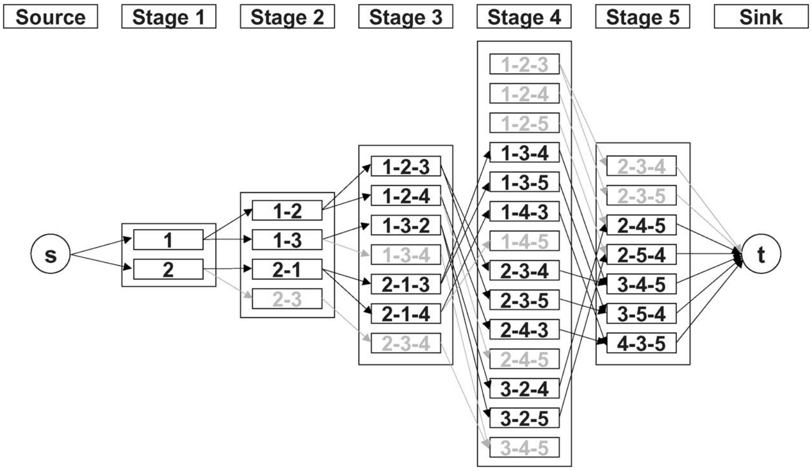


Fig. 1. *Network for $n = 5, k = 1$.*

Some nodes that violate precedence constraints or are not part of a path from source to sink are removed from the network. These nodes are shown in gray in Fig. 1. By this process, we can produce a "pruned" network, which is significantly smaller than the original network. Precedence constraints further reduce the size of the network.

In prior work [16], the following properties of this network were proved.

i) Every possible k-CPS subsequence of length $2k+1$ or less is contained in some node of the network.

ii) Every feasible sequence (one that satisfies maximum position shift constraints and precedence constraints) can be represented by a path in the network from a node in stage 1 to a node in stage n.

iii) Every path in the network from a node in stage 1 to a node in stage n represents a feasible k-CPS sequence.

B. Bounding the Makespan

Given a set of arriving aircraft, the *makespan* is defined as the arrival time of the last aircraft, or, in other words, the completion time of the landing sequence. For a fixed set of aircraft (the static case), minimizing the makespan is equivalent to maximizing the throughput of the schedule and is desirable from the perspective of system performance. As mentioned earlier, since the schedule with the minimum total landing cost could be different from the schedule with the maximum throughput, a determination of the tradeoffs between throughput and landing costs is needed. A possible approach to determining these tradeoffs, and one adopted in this paper, is to determine the minimum landing cost schedule that can be achieved for every feasible value of the throughput. As a first step, given an FCFS schedule, we first determine a range of feasible values of the throughput. A trivial lower bound on the makespan is the minimum value among the earliest arrival times of all aircraft that could land last in the sequence. Similarly, the maximum value among the latest arrival times of all aircraft that could land last in the sequence would provide an upper bound on the makespan.

C. Minimizing the Total Landing Cost

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

$\ell(x)$ The last aircraft of node x.
$\ell'(x)$ The second to 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 land.
$c_j(t)$ Cost of landing aircraft j at time t.
t_j Scheduled time of arrival (STA) of aircraft j.

Let $W_x(t_j)$ be the minimum value of the sum of landing costs that is accumulated until $l(x)$ lands at time t_j. The objective is to minimize the total landing cost, that is, the sum of landing costs of all aircraft

$$\text{Total landing cost} = \sum_{i=1}^{n} c_i(t_i).$$

For an arc (x,y) in the CPS network, the sum of landing costs from the first aircraft of the sequence to the last aircraft i of node x, $W_x(t_i)$, is used to calculate the sum of landing costs from the first aircraft to the last aircraft j of node y, $W_y(t_j)$ using the following dynamic programming recursion:

$$W_y\left(t_{\ell(y)}\right) = \min_{x\in P(y)}\left\{W_x\left(t_{\ell(x)}\right)\right\} + c_{\ell(y)}\left(t_{\ell(y)}\right),$$
$$\forall\ t_{\ell(y)} \in \mathcal{I}(\ell(y)) : t_{\ell(y)} - t_{\ell(x)} \geq \delta_{\ell(x),\ell(y)}.$$

The proof of correctness of this recursion follows standard techniques for proving the validity of dynamic programming recursions and is presented in the Appendix for completeness.

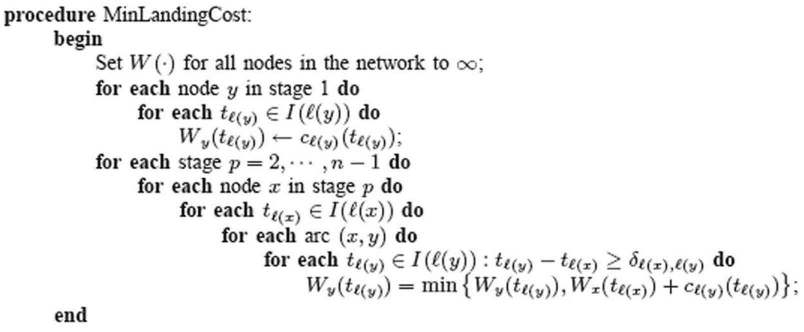
For a node y in the first stage, since there are no previous landing costs, the landing cost is given by $W_y(t_i) = c_i(t_i)$, where i is the last aircraft of the node. For example, i can be 1, 2, or 3 when the maximum number of position shifts allowed $k = 2$.

Since the state space for $W(\cdot)$ is infinite, the recursion is not computationally practical. In order to implement the algorithm, we discretize time into periods of length ϵ. Since radar update rates are once every 10–12 s [25], it would be reasonable to set ϵ to a value between 1 and 10 s. The pseudocode for the algorithm is presented in Fig. 2.

The dynamic programming recursion presented above determines the landing 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 x in stage n of $W_x(t_{\ell(x)})$, such that $t_{\ell(x)} = t$. Comparing $W_x(t_{\ell(x)})$ for all nodes x in stage n, we can also determine the sequence and arrival times of aircraft that minimizes the total landing cost of the schedule.

D. Complexity

It was shown in [16] that the number of nodes in the CPS network is $O\left(n(2k+1)^{(2k+1)}\right)$ and the number of arcs is $O\left(n(2k+1)^{(2k+2)}\right)$. In this case, we have to assign the arrival times of aircraft (and therefore the weight) associated with each arc (x,y) in the given time-window. In a given arc, we need to consider all possible landing times for the last aircraft in node x and the last aircraft of the current node y. When we assume that the length of the largest interval of feasible arrival times among all aircraft is



```
procedure MinLandingCost:
    begin
        Set $W(\cdot)$ for all nodes in the network to $\infty$;
        for each node $y$ in stage 1 do
            for each $t_{\ell(y)} \in I(\ell(y))$ do
                $W_y(t_{\ell(y)}) \leftarrow c_{\ell(y)}(t_{\ell(y)})$;
        for each stage $p = 2, \cdots, n-1$ do
            for each node $x$ in stage $p$ do
                for each $t_{\ell(x)} \in I(\ell(x))$ do
                    for each arc $(x, y)$ do
                        for each $t_{\ell(y)} \in I(\ell(y)) : t_{\ell(y)} - t_{\ell(x)} \geq \delta_{\ell(x),\ell(y)}$ do
                            $W_y(t_{\ell(y)}) = \min\{W_y(t_{\ell(y)}), W_x(t_{\ell(x)}) + c_{\ell(y)}(t_{\ell(y)})\}$;
    end
```

Fig. 2. *Algorithm for computing the minimum landing cost.*

(L/ϵ) and the accuracy is ϵ, there are at most (L/ϵ) time-periods in a given arrival time-window. The computational work done per arc in the CPS network is therefore $O((L/\epsilon)^2)$.

Lemma 1: The complexity of the proposed dynamic programming algorithm is $O(n(2k+1)^{(2k+2)}(L/\epsilon)^2)$, where n is the number of aircraft, k is the maximum allowed number of position shifts, L is the largest difference between the latest and earliest arrival times over all aircraft, and ϵ is the desired resolution.

The proposed method is computationally tractable and amenable to real-time implementation because the complexity scales linearly with the number of aircraft and as the square of the largest difference between the latest and earliest arrival times over all aircraft. While the computational complexity is exponential in k, we note that k is small (typically less than or equal to three).

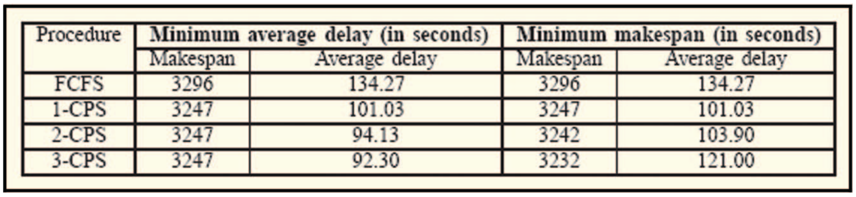
IV. EVALUATING TRADEOFFS BETWEEN OBJECTIVES

A. Minimizing Average Delay

In the previous section, we presented an algorithm for minimizing the sum of landing costs (or equivalently, the sum of delay costs), given the cost of landing each aircraft at a particular time. The problem of minimizing the sum of arrival times of all aircraft can be solved by setting the cost of landing an aircraft at a particular time to be equal to that time (that is, $c_i(t_i) = t_i$). Since the average delay of a given group of aircraft is equal to the sum of the individual delays (differences between the actual and estimated arrival times) divided by the number of aircraft, we can write average delay $= (1/n)(\sum_{i=1}^{n} t_i - \sum_{i=1}^{n} \mathrm{ETA}_i)$. The sum of estimated arrival times is a constant; therefore the problem of minimizing the sum of arrival times is equivalent to that of minimizing the average delay.

We consider a random instance of scheduling a sequence of 30 aircraft on a single runway. The mix of aircraft types is assumed as 40% heavy, 40% large, and 20% small. A discussion of the dependencies of the results on the arrival rates and the fleet mix is beyond the scope of this paper and can be found in [16]. We choose parameters that maintain pressure on the arrival runway (about an aircraft a minute) and a reasonably heterogeneous mix of aircraft. Precedence constraints are imposed by not allowing overtaking between aircraft arriving on the same flight jet route, which is assigned to be one of four possible routes. The earliest arrival time is equal to the ETA $(E(i) = \mathrm{ETA}_i)$, and the maximum allowed delay is 60 min. Table 2 shows the *makespan* (the arrival time of the last aircraft in the group) and the *average delay* for the FCFS and CPS sequences. The throughput of the schedule is given by the number of aircraft in the sequence divided by the makespan of the schedule. When the objective is to minimize average delay (columns 2 and 3 in Table 2), we note that as the maximum number of position shifts k increases, the average delay decreases. We also compare the schedule that minimizes the average delay with the one that minimizes the makespan (using the CPS framework [16]). For each value of k, for the minimum value of the

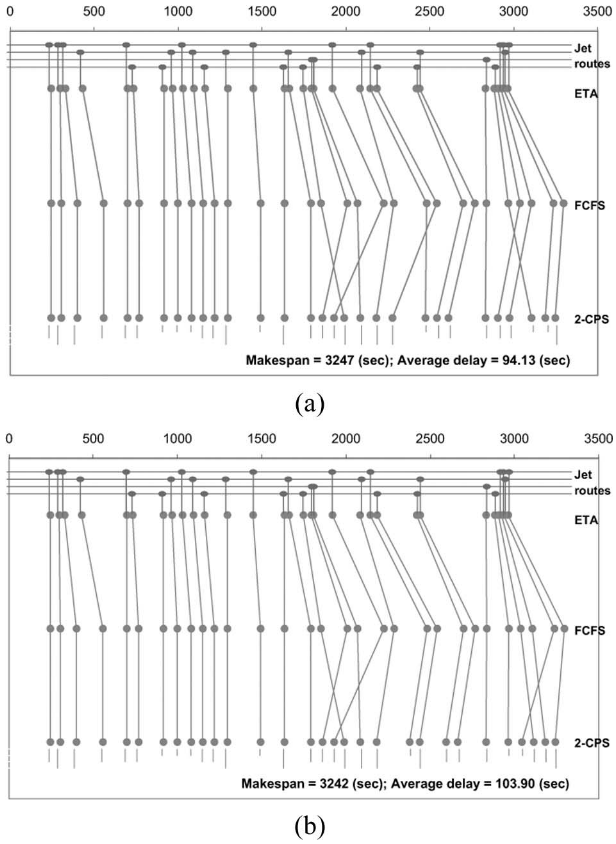
Table 2 Comparison of the Makespan and Average Delay of Various Scheduling Procedures for Two Objectives: 1) Minimizing Average Delay and 2) Minimizing the Makespan



Procedure	Minimum average delay (in seconds)		Minimum makespan (in seconds)	
	Makespan	Average delay	Makespan	Average delay
FCFS	3296	134.27	3296	134.27
1-CPS	3247	101.03	3247	101.03
2-CPS	3247	94.13	3242	103.90
3-CPS	3247	92.30	3232	121.00

makespan, we determine the minimum achievable value of the average delay (shown in Table 2). We note that the decrease in makespan (increase in throughput) is achieved at the cost of an increase in the average delay. While it is true that minimizing the makespan frequently results in an improvement in the average delay [16], this is not necessarily the case. Similarly, minimizing the average delay may result in an increase in makespan (or, equivalently, a decrease in throughput).

The minimum average delay and minimum makespan schedules for $k = 2$ are shown in Fig. 3 in a form popularized in [17], known as the "comb diagram." In Fig. 3, the horizontal lines on the top represent time-lines for each jet route. The dots on each horizontal time-line show when an aircraft is crossing the Center boundary on a given jet route. The time-scales for ETAs and STAs (FCFS and 2-CPS in this instance) have been shifted by a constant amount to make the figure more compact, with the assumption that all aircraft take the same amount of time to travel from the Center boundary to the runway. This can be easily extended to the case where the different travel times are known from the Trajectory Synthesizer [16], [23]. The time-scale given above the comb diagram is for the time-lines of ETAs and STAs. A straight line from a given jet route is connected to the ETA. This time represents the time the aircraft would arrive at the runway if there were no interference from any other aircraft or from unknown navigation errors and environmental conditions. The sequence of all ETAs determines the FCFS order to be preserved for fair scheduling. The horizontal component of the line between ETA and FCFS in the diagram represents the sequenced delay to meet spacing requirements. If the line connecting the ETA of an aircraft to the FCFS schedule is vertical, no delay is required for that particular aircraft; the greater the deviation of the line from the vertical, the more the assigned delay. The crossing of lines connecting the FCFS and 2-CPS schedules denotes the resequencing or exchange of aircraft positions. The vertical line beneath each aircraft on the 2-CPS schedule indicates the weight class of the aircraft: a long line denotes heavy, a medium line denotes large, and a short line denotes small [18]. The makespan of the sequence and the average delay per aircraft in seconds are shown at the bottom of the diagram for each objective function. The two comb diagrams in Fig. 3 illustrate that the position swaps and arrival times in the optimal schedule depend on the objective function and can yield different values of makespan and average delays.



(a)

(b)

Fig. 3. ***Simulated arrival traffic for (a) minimum average delay and (b) minimum makespan, with 2-CPS. The horizontal axes denote the time-line. The ETAs correspond to the estimated time of arrival at the airport if the aircraft flies at its nominal speed and route.***

B. Analysis of Tradeoffs Between Delay and Throughput

We further investigate the tradeoff between average delay and throughput that was demonstrated in the previous section using Monte Carlo simulations. We generate 1000 instances of 30-aircraft sequences, with the aircraft types and jet routes assigned randomly using appropriate probability distributions. Precedence constraints are imposed among aircraft using the same jet route, and time-windows are assigned with the ETA as the earliest arrival time and a maximum delay of 60 min. For each of these generated instances, we optimize the schedule for two different objectives: minimizing the average delay and minimizing the makespan (or maximizing the throughput).

The comparison between the two solutions is shown in Fig. 4(a). The horizontal axis corresponds to the maximum throughput solution and shows its normalized improvement in throughput (CPS throughput − FCFS throughput)/(FCFS throughput). The vertical axis corresponds to the minimum average delay solution and shows its normalized decrease in average delay, which is calculated as (FCFS avg. delay − CPS avg. delay)/(FCFS avg. delay).

We note that about 45% of the instances in Fig. 4(a) lie on the vertical axis. This means that in 45% of the instances, there is little or no benefit (over the FCFS schedule) in minimizing the throughput of the sequence, although there are instances in which a 14% improvement in throughput can be achieved through resequencing the arrival sequence. In contrast, as the histogram in Fig. 4(b) shows, larger improvements in average delay (as high as

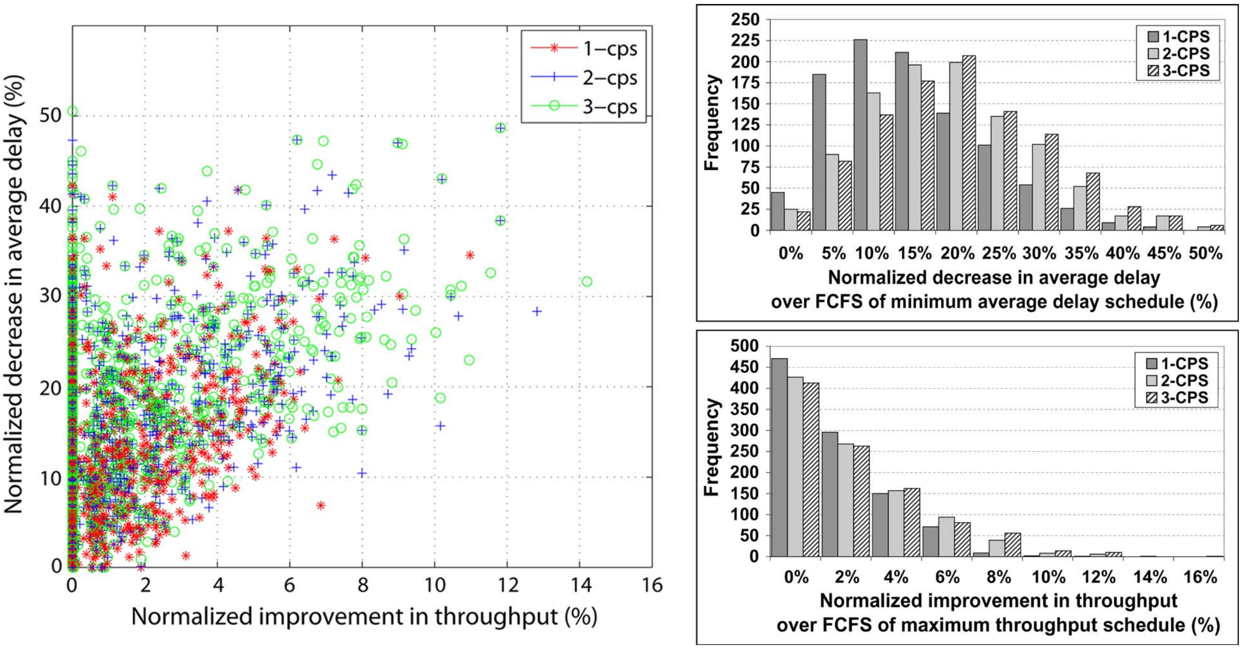


Fig. 4. ***(a) Normalized runway throughput versus normalized average delay. (b) Histograms corresponding to the normalized runway throughput versus normalized average delay solutions.***

50%) can be achieved through resequencing, while the throughput improvements are typically smaller.

Fig. 5(a) shows the makespan values of both the minimum makespan schedule and the minimum average delay schedule. We note the makespan of the schedule that minimizes the average delay does not differ very much from the minimum makespan values. Fig. 5(b) shows the average delay values of both the minimum makespan and the minimum average delay schedules. While in a large number of instances the average delay values are not much larger than the minimum, there are instances in which the average delay corresponding to the minimum makespan solution is significantly greater than the minimum value that can be achieved. In other words, while maximizing the throughput of the sequence, the benefit frequently comes at the expense of an increase in the average delay incurred by the aircraft.

We also compare the minimum average delay and maximum throughput schedules to the nominal FCFS schedules. The rationale behind this is as follows: since the minimum average delay solution can have a suboptimal throughput, it is possible that the throughput of the minimum average delay solution is actually *lower* than the FCFS

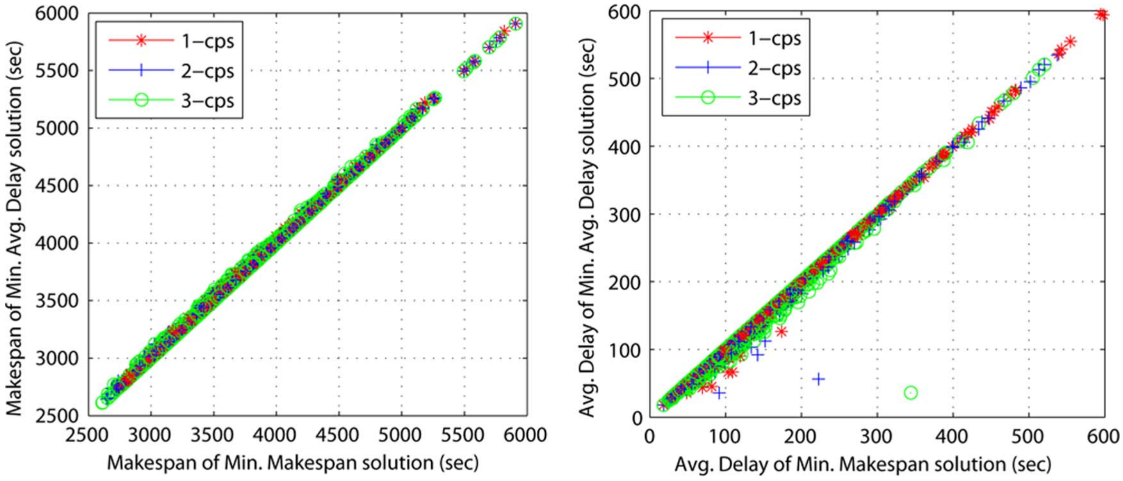


Fig. 5. ***Comparison between the objectives of minimizing average delay and minimizing makespan in terms of (a) makespan and (b) average delay.***

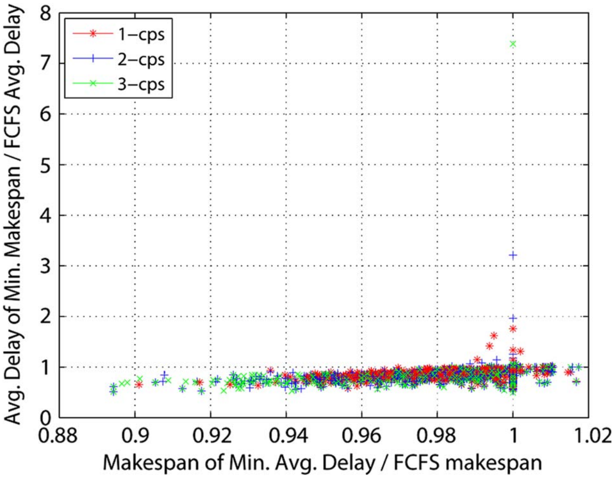
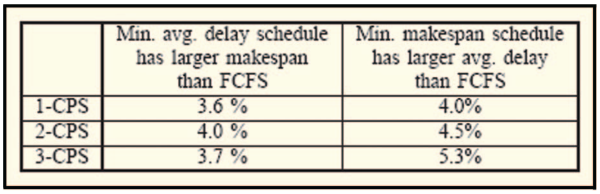


Fig. 6. ***Comparison between optimal and FCFS schedules.***

Table 3 Summary of Comparison Between Optimal and FCFS Schedules



	Min. avg. delay schedule has larger makespan than FCFS	Min. makespan schedule has larger avg. delay than FCFS
1-CPS	3.6 %	4.0%
2-CPS	4.0 %	4.5%
3-CPS	3.7 %	5.3%

throughput. Similarly, the average delay of the minimum makespan solution may be higher than the FCFS average delay. Therefore, for the Monte Carlo simulations, we also compare the ratio of the makespan of the minimum average delay schedule to the FCFS makespan (x-axis in Fig. 6) and the ratio of the average delay of the minimum makespan schedule to the FCFS average delay (y-axis in Fig. 6).

In most samples, these ratios are less than one, that is, the resequencing using CPS improves both the makespan and the average delay when compared to the FCFS solution. However, some instances have a ratio greater than one, implying a worse throughput or average delay than the FCFS schedule. The results are summarized in Table 3. For example, in about 4% of instances, the schedule that minimizes the average delay (with $k = 3$) has a worse throughput than the FCFS schedule. The maximum throughput schedule (with $k = 3$) has a worse average delay than the FCFS schedule in about 5% of instances.

In addition, we note that there are a handful of points that are significant outliers in both Figs. 5 and 6. As we have noted earlier, the minimum makespan solution on average appears to improve the average delay but can sometimes (depending on the FCFS sequence and arrival times) have an adverse effect on the delay. The Monte Carlo simulations show that instances in which the adverse effect is large do occur but are infrequent.

C. Weighted Sum of Delays

While the previous section dealt with the tradeoffs between the average delays and the throughput, it is possible that all the flights may not have equal importance. Some of the possible objectives that could be considered are the passenger-weighted delays or the airline priority weighted delays. We consider an instance similar to the scenarios seen previously for the average delay, but with weighting factors, which are given as one for small aircraft and nine for heavy or large aircraft. The instance includes the estimated arrival times of 30 aircraft, weighting factors of one or nine, and resultant schedules based on FCFS and resequencing to minimize the weighted sum of delays, with a maximum of k position shifts. For each schedule, the total landing cost (sum of delay multiplied by corresponding weighting factors) is computed. As expected, minimizing the weighted sum of delays under CPS tries to land aircraft with large weighting factors as early as possible. In this example, depending on the value of k, resequencing can help save 32–44% of the weighted sum of delays when compared to the FCFS schedule.

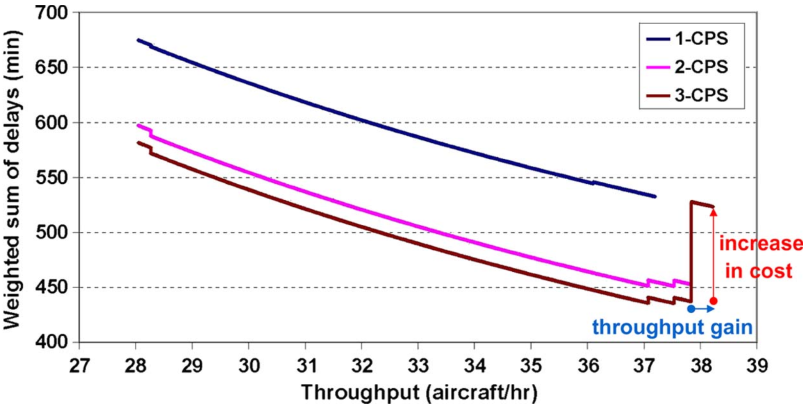


Fig. 7. ***An illustration of the tradeoffs between the weighted sum of delays and the throughput.***

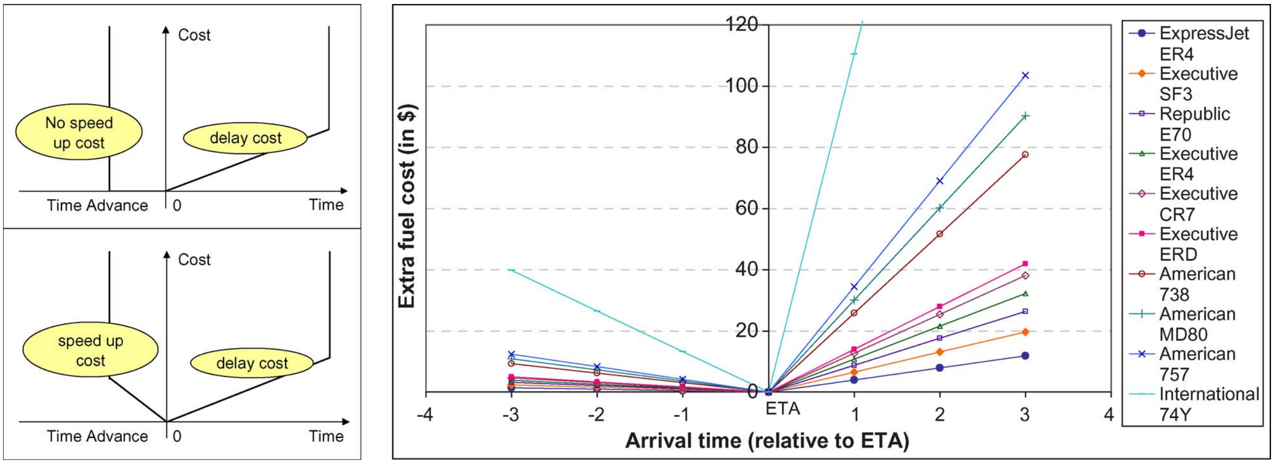


Fig. 8. *(a) Schematic of fuel costs, depending on whether speedup costs are accounted for. (b) Fuel component of landing costs for the top ten aircraft types that operate at DFW.*

In addition, Fig. 7 depicts the relation between the possible throughput and the minimum weighted sum of delays that can be obtained at the throughput for the instance through the *k*-CPS resequencing. In general, as the throughput increases, the weighted sum of delays decreases. However, it is noted that while, by minimizing the makespan, it is possible to increase the throughput from 37.8 aircraft/h to 38.2 aircraft/h, this increase in throughput is achieved at the expense of a 17% increase in the weighted sum of delays.

D. Minimizing Fuel Costs

We apply the proposed algorithm to the problem of minimizing the fuel costs of the arrival schedule at DFW. The airport is located in the Fort Worth Center (ZFW) airspace, and its distance from the Center boundary and other major airports makes it possible to easily determine traffic flow patterns to and from DFW. It is also one of the busiest airports in the United States.

Fuel costs account for almost 50% of the total operating costs per block hour for most airlines [26]. Operating costs, including the cost of fuel consumed per unit delay, are dependent on the specific aircraft types and the airlines. In this paper, we use the latest operating costs based on Form 41 data, in which each airline provides the operating cost breakdown (crew costs, fuel costs, insurance, tax, and maintenance costs per block hour of operation) for each aircraft type that it operates [26]. The fuel costs per unit delay can be derived from this database. A schematic showing the fuel costs, and a graph showing the fuel costs of landing an aircraft at a particular time, are shown in Fig. 8 for the top ten aircraft types that operate at DFW.

Most arrivals into DFW pass through one of four arrival gates before they enter DFW TRACON airspace: BYP (NE gate), CQY (SE gate), JEN (SW gate), and UKW (NW gate) (Fig. 9). Precedence constraints on the landing sequence are imposed based on aircraft that arrive on the same jet route. Runways 18R and 17C are usually used for arrivals. In this paper, it is assumed that runway assignments are decided on the basis of gate usage: since terminals A, C, and E are located on the east side, we assume that all

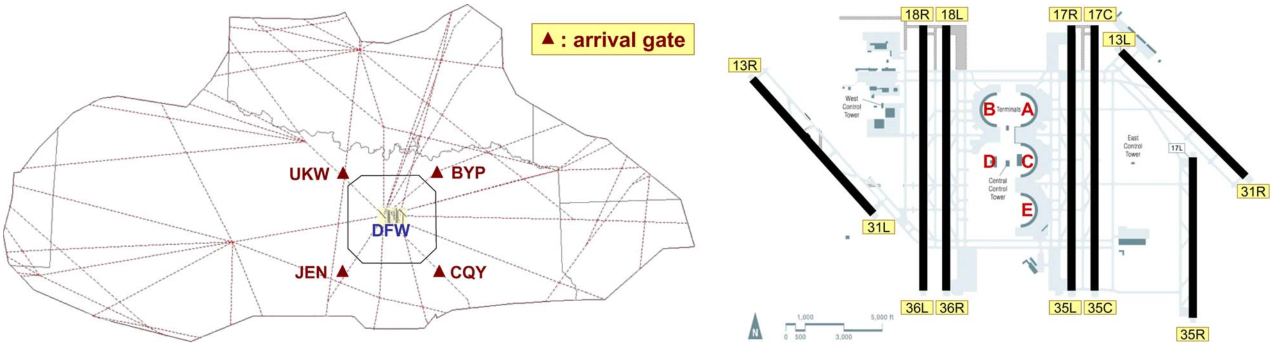


Fig. 9. *(a) The ZFW airspace, showing jet routes and arrival gates. (b) The DFW airport layout, showing runways and terminals [27], [28].*

aircraft using terminal A, C, and E land on runway 17C, and that aircraft using terminals B and D land on 18R. Most of the arrivals at DFW are heavy or large aircraft, with a few small aircraft. It is also assumed that all small aircraft land on 18R, since the southwest area in the terminal is used for general aviation parking. This paper focuses on scheduling arrivals onto runway 18R.

1) *Time Advance:* Neuman and Erzberger noted that if an aircraft was allowed to speed up and land before its ETA, it could potentially result in significant savings in delay for the aircraft that follow it [18]. This procedure of allowing the earliest arrival time $E(i)$ to be less than the ETA is known as *time advance*. However, this decrease in delay (and the associated savings in fuel consumption) is achieved at the expense of the extra fuel that is consumed in speeding up from the nominal velocity profile.

Using the fuel consumption rates, costs, and elapsed times for both the nominal speed profile and the accelerated profile corresponding to various initial speeds and altitudes [18], and calibrating the fuel costs for the nominal profile with the block hour fuel costs of the American Airlines MD80 aircraft (which account for a significant fraction of operations at DFW), the cost per minute of time advance for each airline and aircraft type can be estimated.

The earliest time of arrival is determined by the number of minutes of time advance that is allowed, while the latest time of arrival is chosen such that no aircraft incurs more than 60 min of delay. We consider resequencing with the maximum number of position shifts k varying between 1 and 3 and determine the arrival schedule that minimizes the total fuel cost, accounting for both the fuel cost of delay and that of time advance for each aircraft.

2) *Results:* The ETAs are assumed to be equal to the original scheduled times of arrival, as announced by the airlines [29]. Aircraft are unable to land at the ETAs in practice primarily because of the minimum separation requirements imposed, in addition to the inability to overtake along a jet route. The FCFS landing sequence therefore produces delay and, as a consequence, additional fuel consumption. The data described in the previous sections determine the cost of unit delay and that of unit time advance for each aircraft in the schedule. The extra fuel costs compared to ETAs for scheduling under CPS are calculated and the benefits of the CPS schedule relative to FCFS evaluated.

We consider intervals of 1 h, between 8:00 am and 2:00 pm. Fig. 10 shows extra fuel costs for the different time-windows k and the allowed time advance. As expected, as k increases, the fuel cost savings increase. Similarly, as the allowed speed up increases, the extra fuel cost decreases. However, it is important to note that the marginal benefit decreases, and the curve seems to level off around a value of 3 min time advance. This means that an increase in the cost of fuel required for acceleration begins to offset the fuel-cost benefits of time advance for the rest of the aircraft. It is interesting to note that while the 12:00–1:00 pm and the 1:00–2:00 pm time-windows have the same number of aircraft ($n = 41$), the form of the plot is quite different, with there being little benefit to time advance of more than a minute in the latter case. A closer look at the schedules for the time-windows shows that while the 12:00–1:00 pm window

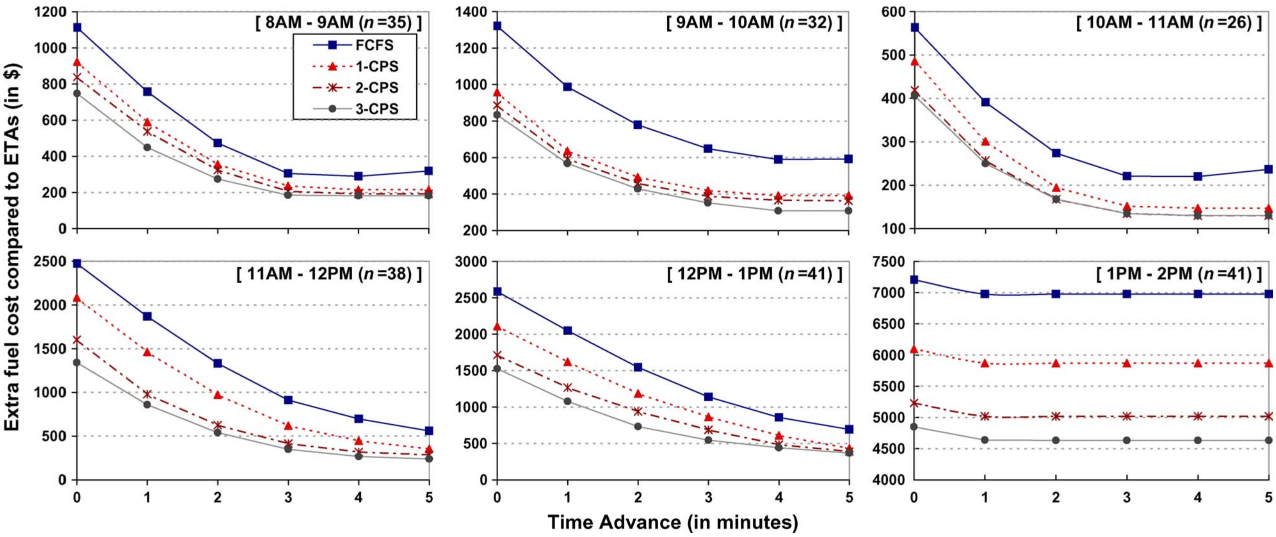


Fig. 10. *Extra fuel cost compared to the ETAs versus the allowed time advance for the minimum fuel cost schedules, for different time-windows.*

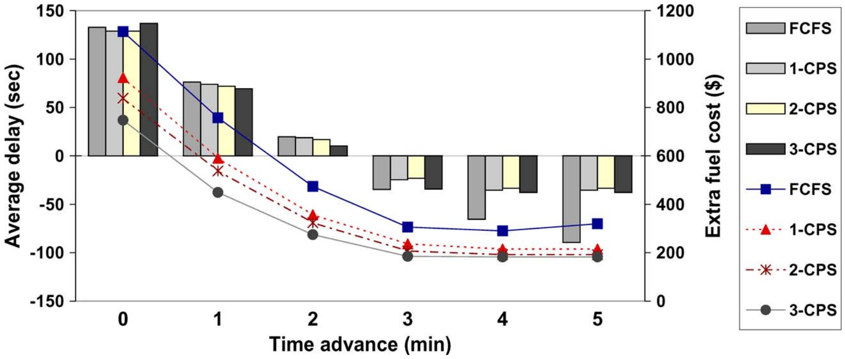


Fig. 11. *Extra fuel cost and average delay versus allowed time advance for the time interval 8:00–9:00 am. The columns show the average delay and the lines show the extra fuel cost incurred.*

has 23 precedence constraints, the 1:00–2:00 pm window has 33 precedence constraints. The heavily constrained sequence prevents aircraft from deriving benefit from time-advance.

The average delay values for the 8:00–9:00 am time-window are shown in Fig. 11. This figure shows that the average delay values do decrease as the amount of time advance increases (this was the primary motivation for time advance). However, for a fixed amount of time advance, the decrease in fuel cost may be achieved at the expense of an increase in average delay.

It is also possible to evaluate the tradeoffs between the minimum fuel cost and maximum throughput objectives, as shown in Fig. 12. The mean decrease in throughput (increase in makespan) experienced due to minimizing the fuel cost is 1.7% of the FCFS throughput (with a standard deviation of 2% and a maximum value of 8%); while when maximizing throughput, the mean increase in fuel cost (over the optimum value) is 1.4% with a standard deviation of 3% and a maximum value of 20%. This suggests that the throughput is on average a (slightly) better objective function since optimizing it only results in a mean increased fuel cost that 1.4% of the FCFS cost; however, considering the worst case scenarios, it is important to note that the maximum increase in fuel cost for the instances seen is 20% compared to a maximum 8% decrease in the throughput.

E. Minimizing Operating Costs

The above experiments are repeated using the total block hour (BH) operating costs reported by the airlines in the Form 41 data, instead of only the fuel costs. The resultant objective functions can be significantly different, as is illustrated in Fig. 13. While the total block hour operating costs of the ERJ145 are comparable for aircraft being operated by either ExpressJet (a regional jet operator based in Houston) or American Eagle, the fuel costs are very different—about 2.7 times as expensive for the latter as for the former. We note that part of this difference can be attributed to the different prices that airlines pay for fuel.

As previously seen for fuel costs in Section IV-D, it is possible to compute the benefit of different amounts of time advance when the total operating costs are being minimized

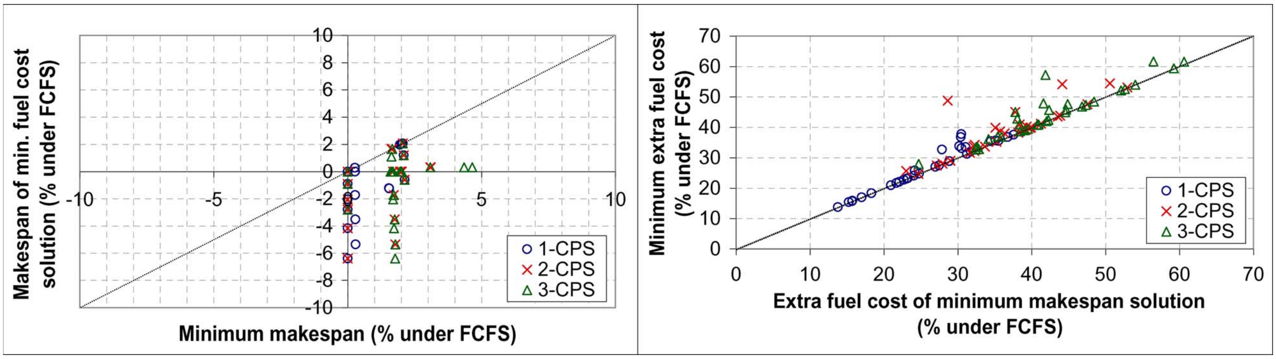


Fig. 12. *Tradeoffs between maximizing throughput and minimizing the fuel costs. The objectives are evaluated relative to the FCFS schedule—for example, the x-axis on the left figure is computed as 100 * (FCFS makespan-min makespan)/ (FCFS makespan).*

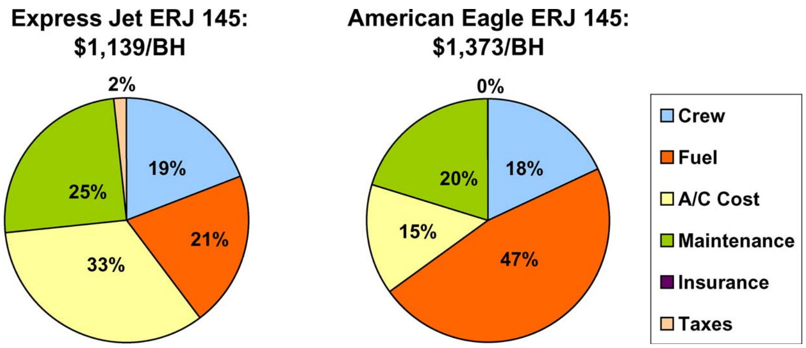


Fig. 13. *Split of the total cost of operating an ERJ145 for two different aircraft operators, ExpressJet and American Eagle. While the total BH operating costs are comparable, the fuel costs are very different.*

(Fig. 14). In contrast to minimizing fuel costs, it appears that there is no point at which the delay benefits in terms of operational costs are offset by the fuel cost of speedup. From the point of view of minimizing total operating costs, it is therefore desirable to allow as much time advance as is practically feasible by aircraft (about 5 min).

The tradeoffs between the total operating cost and throughput objectives are also analyzed, as done previously for average delay versus throughput and fuel cost versus throughput. The results are shown in Fig. 15. This figure shows the tradeoff between the maximum throughput and minimum operating cost solutions. The mean decrease in throughput (suboptimality in the throughput objective) due to minimizing the total operating cost is 0.87% of the FCFS throughput, with a standard deviation of 1.3%, while the mean increase in the operating cost (suboptimality in the operating cost objective) due to maximizing throughput is 0.76% of the FCFS operating cost with a standard deviation of 1.85%. This would suggest that maximizing the throughput on average leads to a (slightly) lower suboptimality in operating cost than the reverse; however, the maximum increase in operating cost for the instances seen is 9% compared to a maximum 4.3% decrease in the throughput.

V. CONCLUSION

This paper presented an algorithm for determining a terminal-area schedule that minimizes the sum of aircraft-dependent delay costs in the presence of constraints such as separation criteria, arrival time windows, limits on deviation from the FCFS sequence, and precedence conditions in a computationally tractable manner. The algorithm was used to optimize schedules and evaluate tradeoffs between different objective functions such as throughput, average delay, fuel costs, and operating costs.

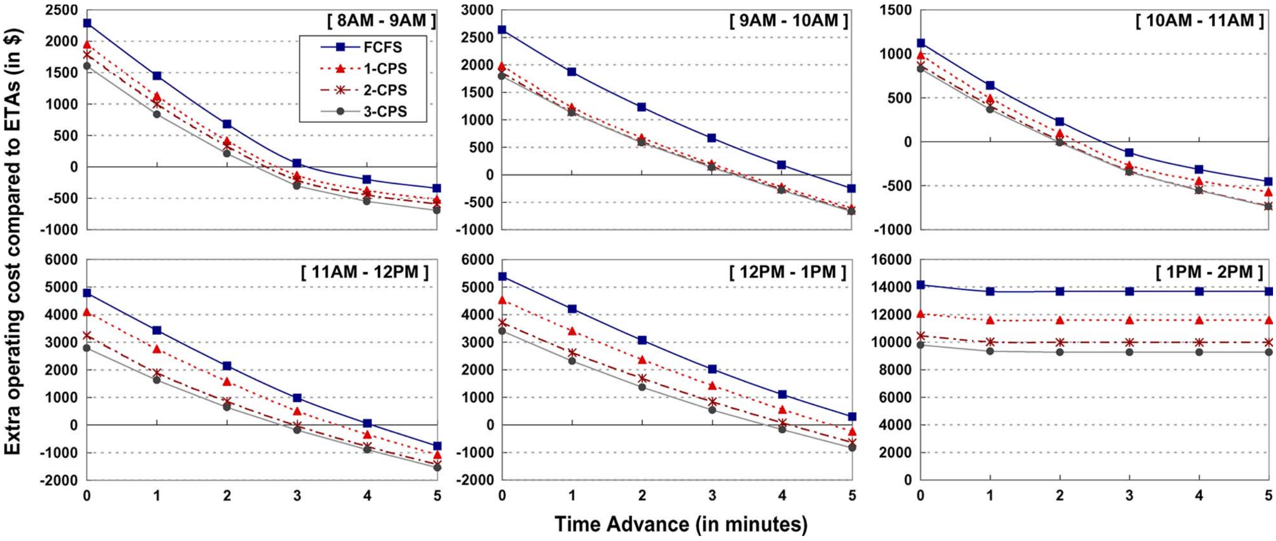


Fig. 14. *Increase in operating costs (over the ETA schedules) versus length of allowed time advance for different 1-h intervals in the day.*

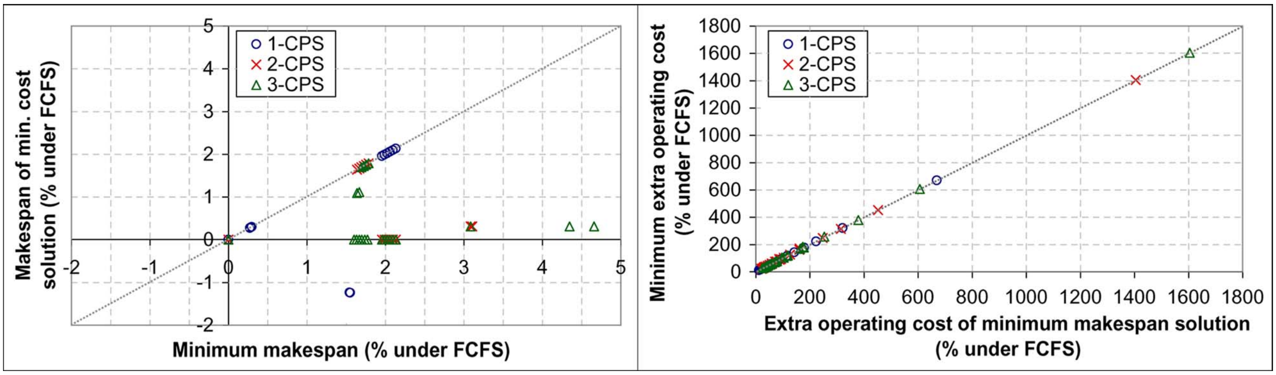


Fig. 15. *Tradeoff between maximizing throughput and minimizing the operating costs.*

The results show that significant improvements in the average delay (up to 50%) can be achieved through resequencing and that the suboptimality of the minimum average delay schedule (measured in terms of the optimal throughput) is quite small. The policy of allowing aircraft to speed up from their nominal profiles and to arrive before their ETAs (known as time advance) was also investigated. The minimum fuel cost schedules were determined by considering, for each aircraft, both the delay cost and the extra fuel cost incurred due to speeding up. The analysis suggested that a time advance of up to 3 min is optimal in most practical scenarios. Using data from Dallas Fort Worth airport, the tradeoffs between fuel and operating costs and runway throughput were also analyzed. The tradeoff analysis showed that minimizing fuel costs or operating costs generally did not result in significant decreases in the throughput of the schedule. It also demonstrated that while on average maximizing the throughput resulted in modest increases in the fuel and operating costs (relative to FCFS), in the worst case scenarios, throughput maximization could result in up to a 20% increase in the fuel cost and up to a 9% increase in the direct operating costs relative to FCFS. The results presented in this paper are an important step toward understanding the tradeoffs between different objectives in terminal-area schedule optimization. ■

APPENDIX

Proof: We first observe that, by construction, $\ell(x) = \ell'(y)$ for $x \in P(y)$. Therefore, $W_y(t_{\ell'(y)}) = W_y(t_{\ell(x)})$. Since $W_y(t_{\ell(y)})$ is the minimum value of total landing cost over all paths leading to node y

$$\begin{aligned} W_y(t_{\ell(y)}) \le\; & W_x(t_{\ell(x)}) + c_{\ell(y)}(t_{\ell(y)}),\\ & \forall\, x \in P(y), t_{\ell(x)} \in \mathcal{I}(\ell(x)), t_{\ell(y)} \in \mathcal{I}(\ell(y)),\\ & \text{where } t_{\ell(y)} - t_{\ell(x)} \ge \delta_{\ell(x),\ell(y)}. \end{aligned}$$

This means that, in particular

$$\begin{aligned} W_y(t_{\ell(y)}) \le\; & \min\nolimits_{x\in P(y)}\{W_x(t_{\ell(x)})\} + c_{\ell(y)}(t_{\ell(y)}),\\ & \forall\, x \in P(y), t_{\ell(x)} \in \mathcal{I}(\ell(x)), t_{\ell(y)} \in \mathcal{I}(\ell(y)),\\ & \text{where } t_{\ell(y)} - t_{\ell(x)} \ge \delta_{\ell(x),\ell(y)}. \end{aligned}$$

To complete the proof, we only need to show that the above relationship can never hold as a strict inequality. For contradiction, suppose that

$$\begin{aligned} W_y(t_{\ell(y)}) < W_x(t_{\ell(x)}) + c_{\ell(y)}(t_{\ell(y)}),&\\ \forall x \in P(y), t_{\ell(x)} \in \mathcal{I}(\ell(x)), t_{\ell(y)} \in \mathcal{I}(\ell(y)).& \end{aligned}$$

This is equivalent to

$$\begin{aligned} & W_y(t_{\ell(y)}) - c_{\ell(y)}(t_{\ell(y)}) < W_x(t_{\ell(x)}),\\ & \quad \forall x \in P(y), t_{\ell(x)} \in \mathcal{I}(\ell(x)), \text{ and } t_{\ell(y)} \in \mathcal{I}(\ell(y))\\ & \Longrightarrow W_y(t_{\ell(y)}) - c_{\ell(y)}(t_{\ell(y)}) < \min_{\substack{z\in P(y)\\ t_{\ell(z)}\in\mathcal{I}(\ell(z))}} \{W_z(t_{\ell(z)})\}\\ & \quad \forall\, t_{\ell(y)} \in \mathcal{I}(\ell(y)). \end{aligned}$$

However, $W_y(t_{\ell(y)}) - c_{\ell(y)}(t_{\ell(y)})$ is the total landing cost of the subsequence of $W_z(t_{\ell(z)})$ that ends at node z and time $t_{\ell(z)}$. This contradicts the minimality of $W_x(t_{\ell(x)})$ for $x = z$. ■

Acknowledgment

The authors would also like to thank the three anonymous referees for their insightful comments and suggestions for improving the readability of this paper.

REFERENCES

[1] R. de Neufville and A. Odoni, *Airport Systems: Planning, Design and Management*. New York: McGraw-Hill, 2003.

[2] U.S. Dept. of Transportation, FY 2004 performance plan, Feb. 2003.

[3] Joint Planning and Development Office (JPDO), Concept of operations for the Next Generation Air Transportation System, Feb. 2007, ver. 1.2.

[4] Joint Planning and Development Office (JPDO). (2004, Dec.). Next Generation Air Transportation System integrated plan. [Online]. Available: http://www.jpdo.gov/

[5] S. Atkins and C. Brinton, "Concept description and development plan for the surface management system," *J. Air Traffic Contr.*, vol. 44, no. 1, Jan.–Mar. 2002.

[6] D. Böhme, "Tactical departure management with the Eurocontrol/DLR DMAN," in *Proc. 6th USA/Eur. Air Traffic Manage. R&D Seminar*, Baltimore, MD, 2005.

[7] F. R. Carr, "Robust decision-support tools for airport surface traffic," Ph.D. dissertation, Massachusetts Inst. of Technology, Cambridge, 2004.

[8] 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 *Proc. 3rd USA/Eur. Air Traffic Manage. R&D Seminar*, Napoli, Italy, Jun. 2000.

[9] J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson, "Scheduling aircraft landings—The static case," *Transp. Sci.*, vol. 34, no. 2, pp. 180–197, 2000.

[10] 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 *Proc. Digital Avion. Syst. Conf.*, 2002.

[11] I. Anagnostakis, J.-P. Clarke, D. Böhme, and U. Völckers, "Runway operations planning and control: Sequencing and scheduling," *J. Aircraft*, vol. 38, no. 6, 2001.

[12] R. G. Dear, "The dynamic scheduling of aircraft in the near terminal area," Ph.D. dissertation, Massachusetts Inst. of Technology, 1976, Flight Transportation Lab. Rep. R76-9.

[13] R. G. Dear and Y. S. Sherif, "An algorithm for computer assisted sequencing and scheduling of terminal area operations," *Transp. Res. A, Policy Practice*, vol. 25, pp. 129–139, 1991.

[14] H. N. Psaraftis, "A dynamic programming approach for sequencing groups of identical jobs," *Oper. Res.*, vol. 28, pp. 1347–1359, 1980.

[15] D. A. Trivizas, "Optimal scheduling with maximum position shift (MPS) constraints: A runway scheduling application," *J. Navig.*, vol. 51, no. 2, pp. 250–266, May 1998.

[16] H. Balakrishnan and B. Chandran, "Scheduling aircraft landings under constrained position shifting," in *Proc. AIAA Guidance, Navig., Contr. Conf. Exhib.*, Keystone, CO, 2006.

[17] F. Neuman and H. Erzberger, "Analysis of sequencing and scheduling methods for arrival traffic," NASA Tech. Memo. 102795, Apr. 1990.

[18] F. Neuman and H. Erzberger, "Analysis of delay reducing and fuel saving sequencing and spacing algorithms for arrival spacing," NASA Tech. Memo. 103880, Oct. 1991.

[19] G. C. Carr, H. Erzberger, and F. Neuman, "Airline arrival prioritization in sequencing and scheduling," in *Proc. 2nd USA/Eur. Air Traffic Manage. R&D Seminar*, Orlando, FL, Dec. 1998.

[20] C. S. Venkatakrishnan, A. Barnett, and A. R. Odoni, "Landings at Logan airport: Describing and increasing airport capacity," *Transp. Sci.*, vol. 27, no. 3, pp. 211–227, 1993.

[21] B. Chandran and H. Balakrishnan, "A dynamic programming algorithm for robust runway scheduling," in *Proc. Amer. Contr. Conf.*, New York, Jul. 2007.

[22] H. Balakrishnan and B. Chandran, "Efficient and equitable departure scheduling in real-time: New approaches to old problems," in *Proc. USA/Eur. Air Traffic Manage. R&D Seminar*, 2007.

[23] R. Slattery, "Terminal area trajectory synthesis for air traffic control automation," in *Proc. 1995 Amer. Contr. Conf.*, Seattle, WA, Jun. 21–23, 1995.

[24] FAA. (2006, Feb. 16). Air traffic control order 7110.65R [includes change 3 (Appendix A)]. [Online]. Available: http://www.faa.gov/atpubs

[25] H. Erzberger, T. J. Davis, and S. M. Green, "Design of center-TRACON automation system," in *Proc. AGARD Meeting Machine Intell. Air Traffic Manage.*, 1993.

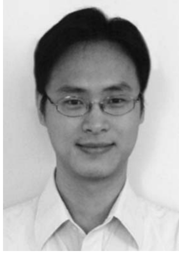
[26] Eclat Consulting, "Aircraft operating costs and statistics," *Aviation Daily*, vol. 368, Jun. 2007.

[27] J. Ott. (2007, Apr.). *Tactical decision aids used by ZFW CWSU*. [Online]. Available: http://www.wrh.noaa.gov/psr/aviation/SAWS_Workshop/25_TDAs_ZFW.ppt

[28] FAA, Airport capacity benchmark report 2004: Dallas/Fort Worth International (DFW). [Online]. Available: http://www.faa.gov/about/office_org/headquarters_offices/ato/publications/bench/2004download.htm

[29] DFW airport arrivals, Jul. 2, 2007. [Online]. Available: http://www.flightstats.com

ABOUT THE AUTHORS



Hanbong Lee (Student Member, IEEE) received the B.S. degree in mechanical and aerospace engineering from Seoul National University, Korea, in 2002. He currently is pursuing the doctoral degree in the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology (MIT), Cambridge.

Prior to joining MIT in 2006, he was with the Korean Air Aerospace Division as a Structural Analysis Engineer in the Boeing 787 wingtip development program. His research interests include air traffic management, optimization algorithms, and the air transportation industry.

Hamsa Balakrishnan (Member, IEEE) received the B.Tech. degree in aerospace engineering from the Indian Institute of Technology, Madras, in 2000 and the Ph.D. degree in aeronautics and astronautics from Stanford University, Stanford, CA, in 2006.

She is the T. Wilson Career Development Assistant Professor of Aeronautics and Astronautics and of Engineering Systems at Massachusetts Institute of Technology (MIT), Cambridge. Prior to joining MIT, she was a Researcher at the University of California, Santa Cruz, and the NASA Ames Research Center. Her research interests address various aspects of air transportation systems, including algorithms for air traffic scheduling and routing, air traffic surveillance algorithms, and mechanisms for the allocation of airport and airspace resources.