

Robust flight schedules through slack re-allocation

Viot Chiraphadhanakul · Cynthia Barnhart

Received: 5 September 2012 / Accepted: 26 April 2013 / Published online: 17 May 2013

© Springer-Verlag Berlin Heidelberg and EURO - The Association of European Operational Research Societies 2013

Abstract A myriad of uncontrollable factors in airline operations makes delays and disruptions unavoidable. To limit complexity, most conventional scheduling models, however, ignore the occurrence of unplanned events during operations. This leads to schedules that are vulnerable to delays and disruptions. In this work, we propose a flight schedule adjustment model that strategically re-allocates existing schedule slack to achieve a more robust schedule. Using data from an international carrier, we evaluate the resulting schedules using various performance metrics, including delay propagation and passenger delays. The results show that minor schedule adjustments to the original schedule can significantly improve overall schedule performance. In addition to the modeling contribution, we provide managerial insights obtained through extensive computational experiments. Unlike earlier works on slack re-allocation that focus primarily on demonstrating the effectiveness of the methods, our paper is, to our knowledge, the first one that also addresses a fundamental question pertaining to how airline characteristics and operations influence the strategy for robust scheduling.

Keywords Robust airline scheduling · Slack re-allocation · Flight scheduling · Passenger delays · Delay propagation

V. Chiraphadhanakul (✉)

MIT Operations Research Center, 77 Massachusetts Avenue Building E40-149, Cambridge,
MA 02139, USA
e-mail: viot@mit.edu

C. Barnhart

MIT School of Engineering, 77 Massachusetts Avenue Room 1-206, Cambridge, MA 02139, USA
e-mail: cbarnhart@mit.edu

Introduction

Delays are inevitable in airline operations due to many unforeseeable factors, such as congested airports, adverse weather conditions, crew sickness, and aircraft mechanical problems. The impact of delays is exacerbated when they propagate to subsequent flights through an airline's interconnected network. Large delays can also lead to flight cancellations and passenger misconnections, causing passengers to wait for several hours for the next available flight.

Flight delays are very costly, and their impacts are far-reaching. According to the recent comprehensive assessment of the costs and impacts of flight delay in the United States by researchers at five universities (Ball et al. 2010), an estimated total cost of \$31.2 billion was incurred due to US air transportation delays in 2007. Specifically, the delays resulted in \$8.3 billion of direct operating costs, including extra fuel consumption and increased crew costs, to airlines. The delay cost to air travelers resulting from flight delays, cancellations, and misconnections was as high as \$16.7 billion. Additionally, extensive flight delays discouraged passengers from traveling by air and led to \$2.2 billion worth of lost demand. Last, the inefficiency of the air transportation industry also resulted in reduced productivity for other businesses and reduced the 2007 US GDP by \$4 billion.

To limit the complexity of airline scheduling approaches, most existing scheduling models assume that every flight will be operated as planned. Ignoring the presence of uncertainties in actual operations results in schedules that are prone to disruptions. In an attempt to reduce delays and airline operating costs, there has been considerable interest recently in building robustness into airline schedules, i.e., proactively creating schedules that are more resilient to delays and disruptions. The key challenge of this problem is to determine a mechanism for improving schedule robustness that can be captured in a tractable mathematical model.

In this work, we investigate a static slack re-allocation approach for robust airline scheduling. Slack is defined as additional time allocated beyond the expected time required for each aircraft connection, passenger connection, or flight leg. To minimize operating costs, airlines have made numerous efforts to increase utilization of all resources in their operations and consequently, to minimize schedule slack. Slack, however, is desirable in robust schedules as it can potentially absorb delays in an airline network, reduce the likelihood of operational disruptions, and provide flexibility to recover once the operation is disrupted. For this reason, a goal of several recent works on robust airline scheduling is to re-allocate, rather than simply increase, existing schedule slack such that the resulting distribution of slack is more effective in absorbing delays and minimizing disruptions, such as flight cancellations and passenger misconnections.

Three major slack re-allocation schemes, namely, aircraft re-routing (Lan et al. 2006; Borndörfer et al. 2010), flight schedule re-timing (Lan et al. 2006; AhmadBeygi et al. 2010), and block time adjustment (Sohoni et al. 2011) have been proposed in the literature. Although the studies each demonstrate the effectiveness of their approaches, one fundamental question remains: how is one slack re-allocation approach more advantageous than another for a given airline with particular characteristics? Many proxies of schedule robustness have also been

proposed and used as objective functions in optimization models; however, there has been little discussion on how different objective functions impact the resulting schedules. More importantly, one common shortcoming of many works is that the robustness of resulting schedules is evaluated based solely on the objective function values as opposed to using multiple evaluation metrics that capture different aspects of complicated airline operations.

The contribution of this paper is a mix of modeling, methodology, and the provision of operational and managerial insights. In particular, we propose a flight schedule adjustment model that strategically re-allocates existing schedule slack to achieve a more robust schedule by adjusting scheduled flight departure and arrival times, while maintaining feasibility of aircraft and passenger connections. Together with the model, we present various objective functions that can potentially lead to more robust schedules. These include an objective function that uses the notion of *effective slack* introduced in this work.

Using data from an international carrier, we present proof-of-concept results where schedule robustness is evaluated based on different performance evaluation metrics including passenger delays and delay propagation. In this work, we use a data-driven, nonparametric approach to solve the optimization problem and evaluate the results. Specifically, we consider each day of operation in historical data as one instance of a delay scenario, rather than modeling delay distributions of each flight independently, to capture correlations of flight delays. Additionally, separate sets of data are used to build robust schedules and evaluate schedule performance as in reality, uncertainty may be observed only partially in the past and/or change over time.

The results show that minor schedule adjustments to the original schedule can significantly improve overall schedule performance. Through comprehensive computational experiments, we discuss model behaviors and address a fundamental question pertaining to how a strategy for robust schedule planning depends on the airline characteristics. The insights provided in this work are useful to both researchers and airlines for selecting an appropriate robust scheduling strategy.

The rest of the paper is structured as follows: In “[Robust airline schedule planning](#)”, we provide background on robust airline schedule planning, including the survey of related literature. In “[Slack re-allocation for robust schedule planning](#)”, we classify different types of slack and demonstrate how an airline can re-allocate schedule slack to minimize delays and disruptions. In “[Robust flight schedule adjustment](#)”, we present the flight schedule adjustment model, together with alternative objective functions. We provide proof-of-concept results in “[Proof-of-concept](#)”, followed by discussions in “[Discussions](#)”. Finally, we conclude our work in “[Conclusion](#)”.

Robust airline schedule planning

The airline schedule planning process

The airline schedule planning process involves considerable complexity with a large number of different airline resources and associated decisions spanning time and geography. The result is an optimization problem which is too large to be solved in a

single step as a single optimization model. Conventionally, the schedule planning process is decomposed into four subproblems: (1) schedule design, (2) fleet assignment, (3) aircraft maintenance routing, and (4) crew scheduling. These subproblems are in fact still so large and complex that optimization models are typically solved deterministically, i.e., assuming that every flight will be operated as planned. For detailed literature reviews of the airline schedule planning process, readers are referred to Belobaba et al. (2009).

Performance evaluation metrics

As in any complex system, there is no single best metric that captures every aspect of an airline's complex operations. Different metrics cannot be used interchangeably. Also, different stakeholders (e.g., airlines, passengers, government) may be interested in different performance metrics. We discuss here three performance metrics on which we will focus extensively in this work.

15-Min on-time arrival performance

A 15-min on-time arrival performance statistic (15-OTP) measures the percentage of flights that arrive at the gate no later than 15 min after the scheduled arrival time as indicated in the Computerized Reservations Systems. It is widely used in the airline industry because it is simple to compute and easy to understand. Additionally, the U.S. Department of Transportation uses 15-OTP to evaluate airline performance and regularly publishes the rankings¹. As a result, many airlines focus on 15-OTP. However, it is not a very good metric for evaluating overall performance of an airline mainly because it does not provide any information about the delay distribution. Given two airlines with the same 15-OTP, one can have a much larger average delay than the other. Moreover, it does not capture the impact of delay propagation in an airline network nor quantify delays experienced by passengers.

Delay propagation

The impact of delays in an airline network can be exacerbated when delays propagate. Because of airline network connectivity, a delay caused by one flight leg can propagate and potentially lead to larger delays on subsequent flight legs. It thus suggests that delay propagation might be a good measure to indicate the robustness of airline schedules. In this section, we present the delay propagation model originally introduced in Lan et al. (2006).

A delay of each flight leg can be decomposed into two components

1. *Propagated delay* occurs when the aircraft to be used for a flight leg is delayed on its preceding flight leg, and there is insufficient slack between the two flights to turn the aircraft. Note that propagated delay is a function of the aircraft routing.

¹ <http://www.dot.gov/airconsumer/air-travel-consumer-reports>.

2. *Independent delay* represents all other delays, such as airborne delay or taxi delay. It is called independent delay because it does not depend on operations of the other flights in the network.

Note that this definition of propagated delay only takes into account the delays due to aircraft arriving late. In reality, a flight may also experience propagated delays caused by delayed crews or passengers.

Figure 1 illustrates the relationship between departures, arrivals, and delays of two flights i and j in the same routing. A solid arrow represents a planned departure time (PDT) and a planned arrival time (PAT) of each flight. A dashed arrow represents an actual departure time (ADT) and an actual arrival time (AAT) of each flight. The time between planned (actual) departure and arrival times is called planned (actual) block time. A planned turn time between flights i and j (PTT_{ij}) is the time between PAT_i and PDT_j . PTT_{ij} must be larger than the minimum turn time (MTT_{ij}) required for turning an aircraft. MTT_{ij} depends on the connection airport, fleet type, and other requirements for flights i and j . The additional time in PTT_{ij} in excess of MTT_{ij} is called (ground time) slack ($Slack_{ij}$).

If the arrival delay of flight i is larger than $Slack_{ij}$, some portion of the delay cannot be absorbed and consequently propagates to flight j . Thus, the total departure delay (TDD) of flight j is comprised of the propagated delay from flight i to flight j (PD_{ij}) and the independent departure delay (IDD) of flight j itself. Similarly, the total arrival delay (TAD) of flight j comprises PD_{ij} and the independent arrival delay (IAD).

Note that IDD captures only the independent delay before a flight is airborne, whereas IAD includes both IDD and the additional independent delay in the air or at the origin and destination airports. One can also view IAD as the difference between the actual block time and the planned block time. More importantly, IDD and IAD may take negative values if an airline expedites the ground process, flies a flight faster, or pads the schedule by increasing the block time to account for potential delays

Mathematically, we have the following relationships:

$$PTT_{ij} = PDT_j - PAT_i \tag{1}$$

$$Slack_{ij} = PTT_{ij} - MTT_{ij} \tag{2}$$

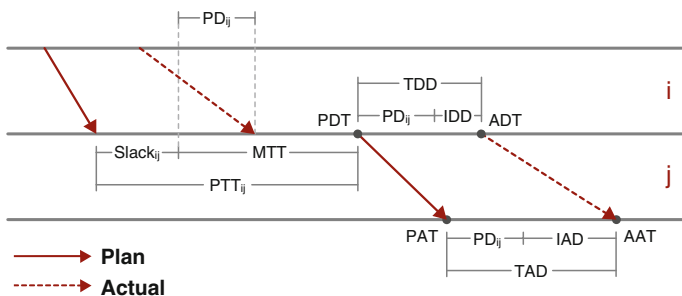


Fig. 1 Flight delay breakdown

$$\text{TDD}_j = \max(\text{ADT}_j - \text{PDT}_j, 0) \quad (3)$$

$$= \max(\text{PD}_{ij} + \text{IDD}_j, 0) \quad (4)$$

$$\text{TAD}_j = \max(\text{AAT}_j - \text{PAT}_j, 0) \quad (5)$$

$$= \max(\text{PD}_{ij} + \text{IAD}_j, 0) \quad (6)$$

$$\text{PD}_{ij} = \max(\text{TAD}_i - \text{Slack}_{ij}, 0). \quad (7)$$

This delay propagation model will be used extensively throughout this work.

Passenger delay

A passenger delay is measured by the difference between the planned arrival time and the actual arrival time at a passenger's final destination. A passenger's itinerary is called *disrupted* if one flight or more in the passenger's itinerary are canceled, or the connection time between consecutive flights becomes less than the minimum connection time required for the passenger to proceed from the respective arrival gate to the subsequent departure gate.

Typically, flight delays underestimate passenger delays because a small flight delay may cause a passenger to mis-connect and then wait for possibly several hours for the next available flight (Bratu and Barnhart 2005). Additionally, flight delay statistics do not capture flight cancellations, which result in many disrupted passengers. Although the number of disrupted passengers might be very small, these disrupted passengers generally represent a large proportion of total passenger delay (Fearing et al. 2010). Consequently, it is increasingly important for airlines to pay attention to passenger delays and make the effort to cut down costs due to passenger re-accommodation and importantly strive to elevate passenger satisfaction.

Robustness in airline scheduling

As mentioned earlier, scheduling models historically have usually ignored the occurrence of unplanned events during actual operations. The results are schedules that are vulnerable to delays and disruptions, thereby incurring higher operational costs than planned. In response, there has been a growing need for robust schedules—ones that are more resilient to delays and disruptions, such as flight cancellations and passenger misconnections. The key challenge of this problem is to determine a mechanism for improving schedule robustness that can be captured in a tractable mathematical model. We summarize here some mechanisms proposed in the literature:

- (a) *Minimizing expected delays and disruptions.* This type of robust schedule minimizes *expected* delays or the *likelihood* of disruptions by means of proxies typically derived from historical data. One possible way to achieve this goal is to maximize schedule slack subject to available resources, for example, by planning aircraft routes with long aircraft connection times or crew pairings with long rest times between duties and long sit times between plane changes (AhmadBeygi et al. 2010; Borndörfer et al. 2010; Burke et al. 2009; Cadarso

- and Ángel Marín, 2013; Dunbar et al. 2012; Eggenberg 2009; Ehrgott and Ryan 2000; Lan et al., 2006; Marla and Barnhart 2010; Schaefer et al. 2005; Sohoni et al, 2011; Yen and Birge 2006).
- (b) *Minimizing the impact of delays and disruptions, once a schedule gets disrupted.* This type of robust schedule is constructed such that, once a schedule gets disrupted, the impact of delays and disruptions is minimal. Two broad ways to achieve this goal are
- *Maximizing recovery flexibility.* Recovery flexibility provides airline operations controllers a recovery solution that requires modest change and is thus relatively inexpensive to execute (Ageeva, 2000; Burke et al. 2009; Eggenberg 2009; Gao et al. 2009; Lapp and Cohn 2012; Rosenberger et al. 2004; Smith and Johnson, 2006; Shebalov and Klabjan, 2006). For example, aircraft swap opportunities (Ageeva, 2000; Burke et al. 2009; Eggenberg 2009) and move-up crews (Shebalov and Klabjan 2006) can be used to prevent further delay propagation.
 - *Isolating delays and disruptions.* A schedule of this kind partitions an airline network into isolated subnetworks such that the delays and disruptions arising in one subnetwork are contained within that subnetwork. Thus, the impact of delays and disruptions is limited, allowing unaffected subnetworks to be operated as planned (Kang 2004; Rosenberger et al. 2004).
- (c) *Minimizing expected schedule costs.* Robustness in an airline schedule usually comes at a cost. This type of robust schedule balances costs of robustness and recovery cost savings (Ageeva 2000; Cadarso and Ángel Marín 2013; Ehrgott and Ryan 2000; Schaefer et al. 2005; Sohoni et al. 2011; Yen and Birge 2006). Typically, this objective is used in conjunction with the aforementioned mechanisms by quantifying monetary values associated with delays, disruptions, or their impacts.

There has been little discussion in the literature on how objective functions might impact schedule performance with respect to different evaluation metrics. In most works, robustness of resulting schedules is evaluated based solely on the objective function values or performance metrics that are directly related to the objective function. Because optimizing one objective may lead to deterioration of another, Burke et al. (2009) consider multi-objective optimization of reliability (on-time probability) and flexibility (swap opportunities). The numerical results exhibit trade-offs between the two objectives, that is, schedules with higher reliability are generally less flexible. Additionally, while increasing flexibility can improve schedule performance, the impact of reliability is dominant. Eggenberg (2009) compares robust schedules created by increasing plane idle time, passenger connection time, and number of plane crossings (hence, swap opportunities). Although the first two objectives are proven to be effective in improving recoverability as well as other delay statistics, increasing the number of plane crossings tends to decrease plane idle time and passenger connection time and does not significantly improve schedule performance.

Methodologies for robust airline schedule planning

In robust schedule planning, historical operations data play an important role in characterizing uncertainty, such as delays or flight durations, in airline networks. Two common approaches for decision making under uncertainty are stochastic programming (Birge and Louveaux 1997; Uryasev and Pardalos 2000; Ruszczyński and Shapiro 2003) and robust optimization (Soyster 1973; Ben-Tal and Nemirovski 1998; Bertsimas and Sim 2004). Both rely on some knowledge of uncertainty.

A stochastic program optimizes the expected value of an objective function involving uncertain data and hence requires full knowledge of underlying distributions of random variables. The distributions may be obtained using a parametric approach, that is fitting empirical data to standard distributions, or a nonparametric approach such as the sample average approximation (SAA) method (Homem-de Mello 2000; Kleywegt et al. 2002; Shapiro 2003) where the expected value of an objective function is approximated by the average over samples. In any case, a certain amount of historical data is required to accurately fit parameters of distributions or approximate the expected value.

Many works in robust airline schedule planning that solve a stochastic optimization problem adopt a parametric approach. For example, Lan (2003) models total arrival delays using a log-normal distribution; Schaefer et al. (2005) model flight delays and ground delays using gamma, Erlang, and beta distributions, depending on flight durations. Because it is difficult to characterize joint distributions of flight delays, these works assume no correlation between flight delays on the same day of operation, which tends to be invalid in reality, especially for those flights departing and arriving at the same airport with the same weather conditions. Additionally, the true distributions of random variables might not be close to any standard parametric distributions. For these reasons, we focus on a nonparametric approach in this work.

A robust optimization problem, on the other hand, optimizes the worst-case scenario and requires only partial knowledge of uncertainty. The main drawback of this approach is that a robust solution is often too conservative as it protects against the worst-case scenario, which is unlikely to happen. Marla and Barnhart (2010) apply the robust optimization framework to the aircraft routing problem and provide extensive analysis of the topic.

Nevertheless, much of research in the literature presents a methodology for improving schedule robustness that does not rely on historical data. In particular, this research exploits domain knowledge and aims at optimizing proxies of robustness that can potentially mitigate the impacts of disruptions, such as aircraft/crew swap opportunities (Ageeva 2000; Shebalov and Klabjan 2006; Burke et al. 2009; Eggenberg 2009) and short cancellation cycle (Rosenberger et al. 2004). Eggenberg et al. (2011) call such proxies of robustness *Uncertainty Features* and propose an *Uncertainty Feature Optimization* framework for optimization under uncertainty without explicitly characterizing randomness. This approach is particularly appealing when a limited amount of historical data is available or it is difficult to characterize uncertainty accurately. In order to determine effective proxies, however, deep domain knowledge is required.

Slack re-allocation for robust schedule planning

Types of slack

We categorize slack in an airline schedule into three types:

1. *Aircraft connection slack (ground time slack)* is additional ground time beyond the minimum turn time of each aircraft connection. The amount of aircraft connection slack in a schedule is a function of an aircraft routing. Aircraft connection slack can be used to absorb accumulated flight delays from prior flights along the aircraft route and thus reduce a likelihood of delay propagation to subsequent flights.
2. *Passenger connection slack* is additional time beyond the minimum connection time between two flight legs in a passenger's itinerary. It is a function of the arrival time of an inbound flight and the departure time of an outbound flight. Passenger connection slack plays an important role in decreasing the chance of passenger misconnection.
3. *Block time slack* is additional time added to the expected block time of each flight. It is a function of a flight's departure and arrival times. Although both block time slack and aircraft connection slack can be used to absorb flight delays, they work differently. Block time slack provides greater flexibility compared with aircraft connection slack. It can absorb propagated delay from the preceding flight, taxi delay (at both departure and arrival airports), and airborne delay, while ground time slack can absorb only propagated delay from the preceding flight.

Slack re-allocation approaches

Despite the advantages of slack in a schedule, from an airline perspective, it is an underutilization of resources. Airlines have made numerous efforts to increase the utilization of all resources in airline operations and consequently, reduce slack in a schedule. Therefore, the recent trend in robust airline scheduling is to re-allocate, rather than simply increase, existing schedule slack. The objective is for the resulting distribution of slack to be more effective in absorbing delays and minimizing disruptions. We summarize here three approaches of static slack re-allocation proposed in the literature.

Figure 2 illustrates how each approach works: a blank outer rectangle denotes a planned flight time of each flight; a filled inner rectangle denotes an actual flight time of each flight, and a thick line connecting two flights represents an aircraft connection. An aircraft connection line is composed of two segments representing minimum turn time and slack. Passenger connections (ZZ 006-ZZ 004 and ZZ 004-ZZ 007) are indicated by thin lines. The shaded regions under ZZ 004 in Fig. 2d and e denote the original flight time.

Aircraft re-routing

In an aircraft re-routing problem (Lan et al. 2006; Borndörfer et al. 2010), the flight schedule and fleet assignment are fixed, i.e., arrival and departure times of every

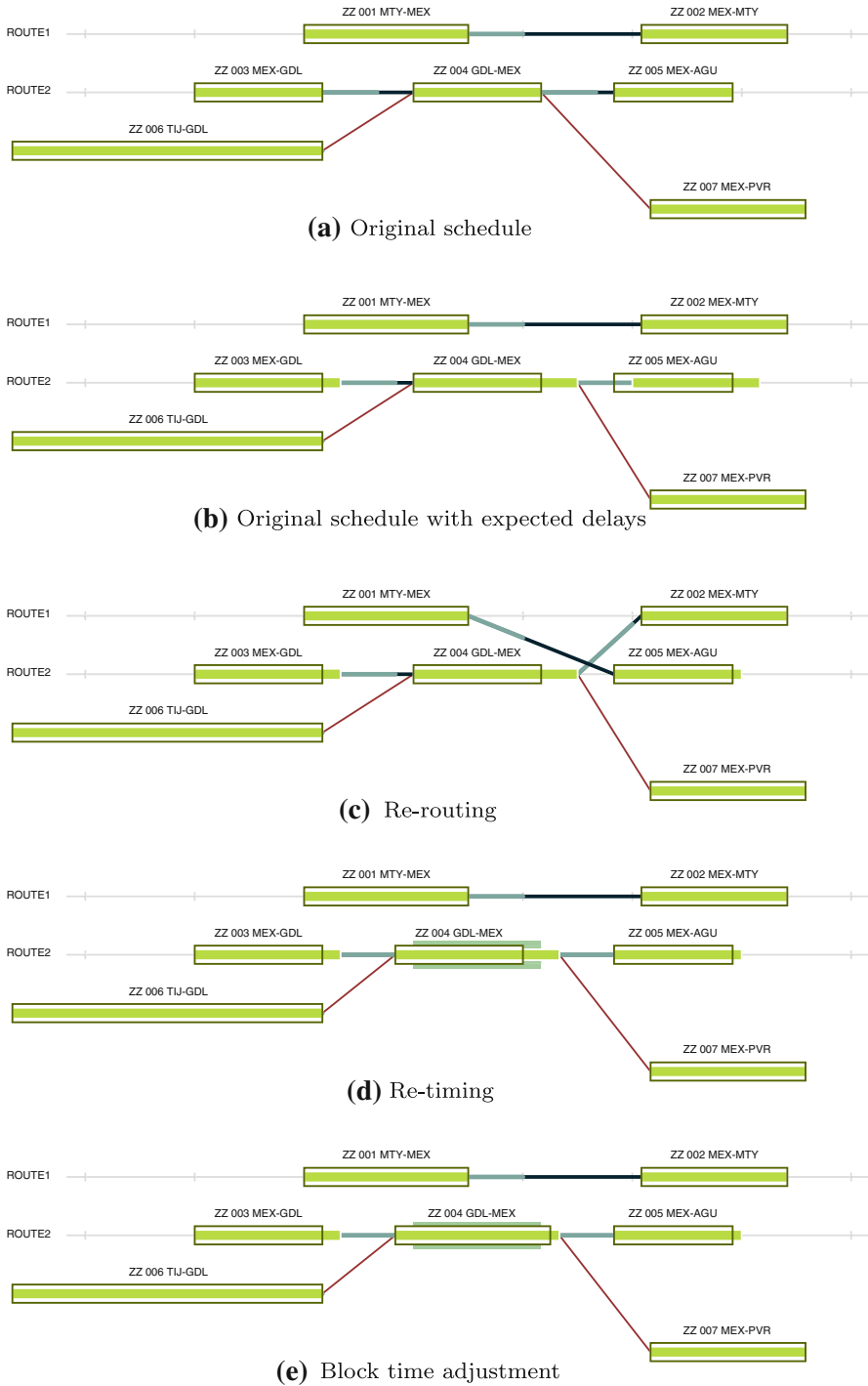


Fig. 2 Slack re-allocation example

flight remain the same as the original schedule, but the aircraft tail assignment of each flight can be changed. As a result, the modified routing yields a different distribution of aircraft connection slack. More ground time slack can be allocated to the connections following flights that are expected to have long arrival delays to prevent delay propagation (see Fig. 2c). Note that the planned passenger connection slack is unaffected as the flight schedule is fixed.

Flight schedule re-timing

In a flight schedule re-timing problem (Lan et al. 2006; AhmadBeygi et al. 2010), aircraft routing and fleet assignment are fixed, but the departure time of each flight is allowed to change within a small time window. The arrival time of each flight must change by the same amount as the departure time, i.e., the block time of each flight is fixed. When a flight is moved earlier, slack in the aircraft connection preceding the flight decreases, whereas slack in the aircraft connection succeeding the flight increases and can potentially absorb its arrival delay (see Fig. 2d). Because a flight schedule is allowed to change, it affects not only aircraft connection slack, but also passenger connection slack.

Block time adjustment

In a block time adjustment problem (Sohoni et al. 2011), aircraft routing and fleet assignment are again fixed, but *both* departure and arrival times of each flight are allowed to change independently. Therefore, in addition to aircraft connection slack and passenger connection slack, it also affects block time slack. In particular, ground time slack can be transformed into block time slack, which can then be used to absorb the flight's independent delay. Figure 2e illustrates how schedule padding helps airlines improve their on-time performance.

Additional detailed examples demonstrating these slack re-allocation approaches are provided in the electronic supplementary materials. While all three approaches will be discussed and compared in the paper, we only present a mathematical model for slack re-allocation through flight schedule adjustment. In particular, the optimization model presented in the next section is capable of solving the flight schedule re-timing problem and the block time adjustment problem. Readers who are interested in the aircraft re-routing problem are referred to Lan et al. (2006) and Chiraphadhanakul (2010).

Robust flight schedule adjustment

Underlying idea

As demonstrated in “[Slack re-allocation approaches](#)”, adjusting flight scheduled departure and arrival times can affect aircraft connection slack, passenger connection slack, and block time slack. Given a set of historical data, we want to strategically re-allocate the existing slack in a schedule by adjusting the flight scheduled departure and arrival times such that the resulting distribution of slack

minimizes some proxy of expected delays or disruptions, while feasibility of aircraft and passenger connections is preserved.

The model

We first introduce the notations used in this formulation:

Sets

- F set of flight legs
- A set of aircraft connections
- P set of passenger connections
- Ω set of possible delay scenarios
- p_ω probability that a delay scenario $\omega \in \Omega$ occurs

Data

- aSlack_{ij} original planned aircraft connection slack in an aircraft connection $(i, j) \in A$
- pSlack_{ij} original planned passenger connection slack in a passenger connection $(i, j) \in P$
- IAD_i^o independent arrival delay of flight $i \in F$ for a given delay scenario $\omega \in \Omega$

Variables

- pd_{ij}^o propagated delay from flight leg $i \in F$ to the succeeding flight leg $j \in F$ for a given delay scenario $\omega \in \Omega$
- tad_i^o total arrival delay of flight leg $i \in F$ for a given delay scenario $\omega \in \Omega$
- aSlack'_{ij} resulting planned aircraft connection slack in an aircraft connection $(i, j) \in A$ after schedule adjustment
- pSlack'_{ij} resulting planned passenger connection slack in a passenger connection $(i, j) \in P$ after schedule adjustment

For each flight $i \in F$, let x_i be the difference between the new and the original *departure* time, and y_i be the difference between the new and the original *arrival* time. The variable $x_i(y_i)$ takes a negative value if the departure(arrival) time is moved earlier and takes a positive value if the departure(arrival) time is moved later. We limit the change in the departure time of each flight i to a small time window $[l_{x_i}, u_{x_i}]$, and the arrival time to $[l_{y_i}, u_{y_i}]$. Additionally, the total change in a block time of each flight is limited within the range $[l_i, u_i]$. Note that the model allows block time reduction of some flights as well.

Using the notions of delay propagation described in “[Delay Propagation](#)”, the robust flight schedule adjustment model is given by

$$\text{Minimize} \quad \sum_{i \in F} \mathbb{E}[\text{tad}_i] = \sum_{i \in F} \left(\sum_{\omega \in \Omega} p_\omega \text{tad}_i^\omega \right) \tag{8}$$

$$\text{subject to } \text{aSlack}'_{ij} = \text{aSlack}_{ij} - y_i + x_j \quad \forall (i,j) \in A \tag{9}$$

$$\text{aSlack}'_{ij} \geq 0 \quad \forall (i,j) \in A \tag{10}$$

$$\text{pSlack}'_{ij} = \text{pSlack}_{ij} - y_i + x_j \quad \forall (i,j) \in P \tag{11}$$

$$\text{pSlack}'_{ij} \geq 0 \quad \forall (i,j) \in P \tag{12}$$

$$\text{pd}^{\omega}_{ij} \geq \text{tad}^{\omega}_i - \text{aSlack}'_{ij} \quad \forall (i,j) \in A, \forall \omega \in \Omega \tag{13}$$

$$\text{pd}^{\omega}_{ij} \geq 0 \quad \forall (i,j) \in A, \forall \omega \in \Omega \tag{14}$$

$$\text{tad}^{\omega}_i \geq \text{IAD}^{\omega}_i + x_i - y_i \quad \forall i \in F_0, \forall \omega \in \Omega \tag{15}$$

$$\text{tad}^{\omega}_j \geq \text{pd}^{\omega}_{ij} + \text{IAD}^{\omega}_j + x_j - y_j \quad \forall (i,j) \in A, \forall \omega \in \Omega \tag{16}$$

$$\text{tad}^{\omega}_i \geq 0 \quad \forall i \in F, \forall \omega \in \Omega \tag{17}$$

$$l_i \leq y_i - x_i \leq u_i \quad \forall i \in F \tag{18}$$

$$l_{x_i} \leq x_i \leq u_{x_i} \quad \forall i \in F \tag{19}$$

$$l_{y_i} \leq y_i \leq u_{y_i} \quad \forall i \in F \tag{20}$$

The objective function (8) is to minimize the total expected arrival delay over all flights. By assuming that Ω has finite cardinality, we have that $\mathbb{E}[\text{tad}_i] = \sum_{\omega \in \Omega} p_{\omega} \text{tad}^{\omega}_i$.

Constraint (9) captures the resulting planned slack of each aircraft connection $(i,j) \in A$ after moving the departure and arrival times of flights i and j . The resulting planned slack (aSlack'_{ij}) increases when the arrival time of flight i is moved earlier, i.e., when y_i takes a negative value. Also, it increases when the departure time of flight j is moved later, i.e., when x_j takes a positive value. The non-negativity constraints of the resulting planned aircraft connection slack, (10), ensure that every aircraft connection is longer than the required minimum turn times and thus remains feasible.

Similarly, constraint (11) captures the resulting planned passenger connection slack for each passenger connection, and the non-negativity constraints of the resulting planned passenger connection slack, (12), ensure that every passenger connection remains feasible.

Given an adjusted flight schedule, the propagated delays for each aircraft connection under different delay scenarios are given by constraints (13) and (14); constraints (15)–(17) determine the total arrival delays for each flight leg under different delay scenarios, assuming that the first flight of each string—a sequence of flights flown by the same aircraft—has zero propagated delay. As discussed earlier, an increase in block times can help absorb independent delays, while a decrease in block times can result in larger independent delays. Therefore, the total arrival delay of each flight changes with changes in block time, and one can think of the term $\text{IAD}^{\omega}_j + x_j - y_j$ in (15) and (16) as the resulting independent arrival delay of flight j after schedule adjustment.

Constraints (18) restrict the total change in block time of each flight within the range $[l_i, u_i]$. Last, constraints (19) and (20) limit the change in the departure and arrival times of each flight i within specific time windows $[l_{x_i}, u_{x_i}]$ and $[l_{y_i}, u_{y_i}]$.

Theorem 1 *The polyhedron formed by constraints (9)–(20) is integral, given that all data and parameters in those constraints are integral.*

The proof of the theorem is provided in the electronic supplementary materials. Because airlines can only schedule flight times on the minute, if all data and parameters (in minutes) are integral, it follows from the theorem that the optimal solution is integral and needs no further rounding.

A special case

We can use the given formulation for solving the flight schedule re-timing problem by enforcing l_i and u_i in constraints (18) to be zeroes. Consequently, we have $x_i = y_i$ for each flight $i \in F$, and the block time remains unchanged.

This particular model can be considered a variation of the flight schedule re-timing formulation proposed by AhmadBeygi et al. (2010). In particular, the decision variables—the changes in the departure times—are modeled in a similar manner, but the calculations of delays are different.

Our model considers only the delay propagation due to aircraft arriving late, whereas their model considers the delay propagation due to aircraft and cockpit crews. As a result, our calculation of the total propagated delay can be accomplished simply through aircraft routes, using the notion of propagated delay introduced in “[Delay Propagation](#)”. AhmadBeygi et al. (2010), on the other hand, propose the notion of a *propagation tree* to capture delay propagation from multiple resources. As mentioned in their paper, their model using propagation trees still cannot accurately take into account simultaneous delays from different propagation trees.

Another major difference is that we allow independent arrival delays to take negative values to reflect overestimated block times of some flights or overestimated minimum turn times of some aircraft connections according to historical data. Last, we also explicitly enforce the feasibility of every existing passenger itinerary.

Alternative objective functions

As discussed in “[Performance evaluation metrics](#)”, there is no single best metric that captures every aspect of an airline’s complicated operations. We present in this section alternative objective functions that aim to minimize delay propagation and passenger delays.

Minimizing total expected propagated delay

In the presented formulation, propagated delays for each given delay scenario $\omega \in \Omega$ are computed and maintained to determine the expected total arrival delay in the objective function. Therefore, it is straightforward to modify the objective function to minimize total expected propagated delay. Specifically, the objective function of minimizing the total expected propagated delay is given by

$$\text{Minimize} \quad \sum_{(i,j) \in A} \mathbb{E}[\text{pd}_{ij}] = \sum_{(i,j) \in A} \left(\sum_{\omega \in \Omega} p_{\omega} \text{pd}_{ij}^{\omega} \right). \tag{21}$$

This objective function is previously presented in Lan et al. (2006), AhmadBeygi et al. (2010).

Maximizing total expected effective slack

Recall that the propagated delay from flight i to flight j takes a positive value only when the total arrival delay of flight i exceeds the planned slack in that connection, as depicted in Fig. 3a. As a result, propagated delay PD_{ij} serves to mask the arrival delay of flight i when $\text{TAD}_i < \text{Slack}_{ij}$. In particular, given two solutions with $\text{PD}_{ij} = 0$ and different TAD_i , the model that minimizes total propagated delay cannot distinguish between the two solutions, even though the one with smaller TAD_i is more desirable.

To overcome this difficulty, we introduce the notion of *effective slack*. Let i and j be two consecutive flights in the same string. We define the effective slack in the connection between flights i and j ($\overline{\text{Slack}}_{ij}$) as

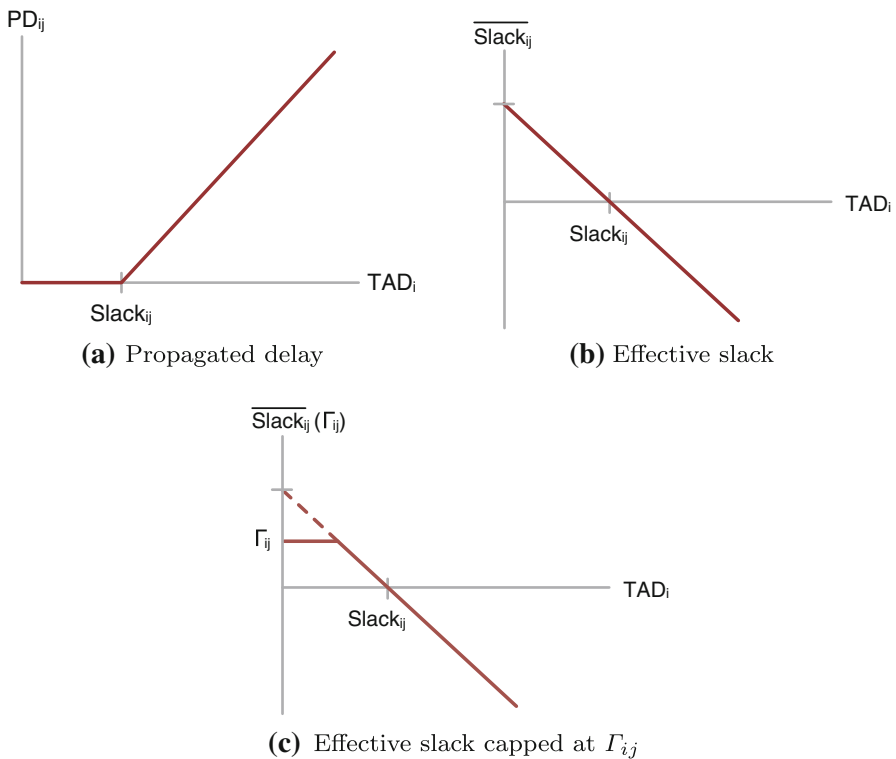


Fig. 3 Propagated delay versus effective slack

$$\overline{\text{Slack}}_{ij} = \text{Slack}_{ij} - \text{TAD}_i. \tag{22}$$

In other words, the effective slack in each connection represents the remaining slack after accounting for the arrival delay of the inbound flight (see Fig. 3b). Note that effective slack may take a negative value. In this case, the arrival delay of the inbound flight will propagate to the outbound flight.

The proposed objective function is to maximize total expected effective slack. To ensure that the model has no incentive to add more slack to connections that already have a reasonable amount of slack, we introduce another parameter—a nonnegative cap Γ_{ij} for each aircraft connection from flight i to flight j . We then redefine the effective slack as the minimum of Γ_{ij} and the difference of the planned slack and the total arrival delay associated with the aircraft connection from flight i to flight j . Specifically,

$$\overline{\text{Slack}}_{ij}(\Gamma_{ij}) = \min(\text{Slack}_{ij} - \text{TAD}_i, \Gamma_{ij}). \tag{23}$$

Therefore, any connection from flight i to flight j with effective slack more than the cap Γ_{ij} contributes only Γ_{ij} minutes to the objective function. This results in the allocation of more slack to connections for which the expected effective slack is smaller than the corresponding cap. An airline can set cap values to reflect how much they are willing to protect against unexpected delay. In addition, because caps are specific to connections, we can set them to different values for different fleet types, connecting airports, and so forth.

We can apply the notion of *effective slack* to aircraft connections in the presented formulation as follows. Let $\overline{\text{aSlack}}_{ij}^\omega$ be the effective aircraft connection slack associated with a connection $(i, j) \in A$ capped at a nonnegative level Γ_{ij} for a given delay scenario $\omega \in \Omega$. Because the resulting planned aircraft connection slack of the aircraft connection (i, j) after schedule adjustment is given by aSlack'_{ij} , we have

$$\overline{\text{aSlack}}_{ij}^\omega = \min(\text{aSlack}'_{ij} - \text{tad}_i^\omega, \Gamma_{ij}). \tag{24}$$

Therefore, the flight schedule adjustment model maximizing the total expected effective aircraft connection slack is given by

$$\text{Maximize} \quad \sum_{(i,j) \in A} \left(\sum_{\omega \in \Omega} p_\omega \overline{\text{aSlack}}_{ij}^\omega \right) \tag{25}$$

$$\text{subject to} \quad \overline{\text{aSlack}}_{ij}^\omega \leq \text{aSlack}'_{ij} - \text{tad}_i^\omega \quad \forall (i, j) \in A \tag{26}$$

$$\overline{\text{aSlack}}_{ij}^\omega \leq \Gamma_{ij} \quad \forall (i, j) \in A \tag{27}$$

(9)-(20).

In fact, the objective function of minimizing total expected propagated delay in (21) is a special case of the proposed objective function with caps set equal to zero for all aircraft connections. In particular, for any aircraft connection from flight i to flight j , we have

$$\begin{aligned} \overline{\text{aSlack}}_{ij}^\omega(\Gamma_{ij} = 0) &= \min(\text{aSlack}'_{ij} - \text{tad}_i^\omega, 0) = -\max(\text{tad}_i^\omega - \text{aSlack}'_{ij}, 0) \\ &= -\text{pd}_{ij}^\omega. \end{aligned}$$

Moreover, one can also apply the notion of *effective slack* to passenger connections to provide sufficient slack in passenger connections where needed and reduce the likelihood of passenger misconnection. Similarly to aircraft connections, we have $\overline{\text{pSlack}}_{ij}^\omega$, the effective passenger connection slack associated with a connection $(i, j) \in P$ capped at a nonnegative level Γ_{ij} for a given delay scenario $\omega \in \Omega$, is defined as

$$\overline{\text{pSlack}}_{ij}^\omega = \min(\text{pSlack}'_{ij} - \text{tad}_i^\omega, \Gamma_{ij}), \tag{28}$$

and the flight schedule adjustment model maximizing the total expected effective passenger connection slack is given by

$$\text{Maximize} \quad \sum_{(i,j) \in P} \left(\sum_{\omega \in \Omega} p_\omega \overline{\text{pSlack}}_{ij}^\omega \right) \tag{29}$$

$$\text{subject to} \quad \overline{\text{pSlack}}_{ij}^\omega \leq \text{pSlack}'_{ij} - \text{tad}_i^\omega \quad \forall (i, j) \in P \tag{30}$$

$$\overline{\text{pSlack}}_{ij}^\omega \leq \Gamma_{ij} \quad \forall (i, j) \in P \tag{31}$$

(9)-(20).

Proof-of-concept

Data and evaluation process

We obtained 3 months of historical operations data (from January 1st to March 25th 2008) from an international carrier. The dataset contains flight information (origin, destination, scheduled/actual departure, and arrival times), planned aircraft routing, minimum turn times currently used by the airline, and passenger booking data (for March data only). Because airlines use historical data to build schedules for future operations, we divide our dataset into two disjoint subsets representing historical data and future operations.

Algorithm 1 Obtaining Historical Independent Arrival Delays

1. For each pair of consecutive flight legs i and j flown by the same aircraft in delay scenario ω , the propagated delay from flight i to flight j is given by $PD_{ij}^\omega = \max(TAD_i^\omega - aSlack_{ij}, 0)$, where TAD_i^ω is the total arrival delay of flight i in the historical data, and $aSlack_{ij}$ is the aircraft connection slack between flights i and j in the original schedule.
 2. For each flight leg j ,
 - if j is the first flight of a string, $IAD_j^\omega = TAD_j^\omega$;
 - otherwise, $IAD_j^\omega = TAD_j^\omega - PD_{ij}^\omega$.
-

In this work, we focus on a data-driven, nonparametric approach. In particular, we use the sample average approximation method and consider each day of operation in January and February as one instance of a delay scenario ω , assuming

that each delay scenario is equally likely. Hence, the set of delay scenarios Ω has cardinality $31 + 29 = 60$. For each day of operation, we can obtain historical independent arrival delays using Algorithm 1, which assumes that the first flight of each aircraft route has zero propagated delay. Note that flight legs in different scenarios are matched using flight numbers and departure airports. When a flight is not included in some scenarios, we use the average historical independent arrival delay over the scenarios containing the flight. We then solve the proposed robust flight schedule adjustment model over each day of operation to obtain a planned schedule for March. The size of a problem on each day of operation is summarized in Table 1.

Next, we use the actual delay information from March data to evaluate performance of the robust schedules. Specifically, we again apply Algorithm 1 to compute the actual independent arrival delays of every flight for each day of operation in March. Given the actual independent delay of each flight, we simulate the actual departure and arrival times of each flight with respect to the new

Table 1 Problem sizes

Day of operation (March 2008)	Number of flights	Number of aircraft	Number of passenger connections	Number of passengers
1	231	59	469	12,840
2	243	61	544	16,027
3	250	58	520	13,889
4	236	57	449	12,455
5	240	59	514	14,187
6	260	59	560	15,288
7	263	61	595	17,416
8	242	61	506	14,620
9	243	60	566	16,485
10	246	59	464	14,634
11	226	55	299	8,725
12	232	57	461	16,258
13	268	61	561	19,891
14	260	62	606	22,906
15	242	61	470	19,871
16	246	61	523	18,040
17	247	61	511	15,192
18	229	57	409	12,918
19	230	59	496	14,718
20	231	60	500	13,993
21	211	58	447	11,409
22	249	62	430	16,192
23	244	61	557	20,648
24	252	61	583	19,226
25	229	56	453	15,201

schedules, assuming no real-time recovery, such as flight cancellations and aircraft swaps, during the day of operation.

To our knowledge, Lan et al. (2006) is the only work in the airline literature that also uses separate sets of historical data to model uncertainty for optimization and evaluation. Many papers in the literature entail building robust schedules and evaluating schedule performance using the same distributions of random variables. Consequently, the evaluation fails to account for the fact that uncertainty may be observed only partially before planning and/or change over time.

Passenger delay calculation

As discussed in “[Passenger delay](#)”, understanding the extent of passenger delays is crucial in evaluating schedule performance. In this work, we calculate passenger delays using the Passenger Delay Calculator algorithm (PDC) developed by Bratu and Barnhart (2005). The algorithm first determines, according to the actual flight operation data, if an itinerary is *disrupted*, that is, one or more flights in the itinerary are canceled, or connecting time between a pair consecutive flights becomes less than the minimum connecting time required. Non-disrupted passengers are assigned to their original itinerary, whereas each disrupted passenger is reassigned, on a first-disrupted-first-recovered basis, to the best recovery itinerary—the one that arrives earliest at his or her final destination and contains at least one available seat.

The passenger delay of a non-disrupted passenger is given by the arrival delay of the last flight in his or her itinerary. For each disrupted passenger, the passenger delay is the difference between the planned arrival time of the last flight in his or her original itinerary and the actual arrival time of the last flight in his or her recovery itinerary. If the passenger delay associated with a disrupted passenger exceeds 12 h, it is assumed that the passenger, referred to as *spilled passenger*, is re-accommodated on another airline and experiences delay of 12 h (Because flight schedules of other airlines as well as seat availability information are not available, we cannot accurately compute passenger delays for disrupted passengers that are re-accommodated on other airlines). Last, for simplicity, we consider only itineraries with at most two flight legs (i.e., at most one connection). According to the dataset, almost 99 % of passengers are represented by such itineraries.

For details of the PDC algorithm and discussion on the validity of the assumptions underlying it, readers are referred to Bratu (2003) and Bratu and Barnhart (2005).

Computational results

In this section, we present the computational results obtained from the robust flight schedule adjustment model introduced in “[Robust flight schedule adjustment](#)”, with different objective functions and parameters. The optimization model was implemented using Java 1.5 and IBM ILOG CPLEX 11.2. For each instance, the optimal solution (a 25-day schedule for March) could be obtained within 5 min on a 2-GHz Intel Core 2 Duo machine with 2 GB of RAM.

Let *Original* denote the airline’s original schedule, and define the solutions to the model with different objective functions as follows:

minTAD minimizes the total expected arrival delay (see (8))

minPD minimizes the total expected propagated delay (see (21))

maxEffACSlack $\langle \Gamma \rangle$ maximizes the total expected effective *aircraft connection* slack with caps set equal to Γ minutes for every aircraft connection (see (25))

maxEffPaxSlack $\langle \Gamma \rangle$ maximizes the total expected effective *passenger connection* slack with caps set equal to Γ minutes for every passenger connection (see (29))

Recall that in the model, we need to specify six parameters $(l_{x_i}, u_{x_i}, l_{y_i}, u_{y_i}, l_i,$ and $u_i)$ that limit the allowable changes in the departure time, arrival time, and block time of a given flight i . Throughout this section, we assume, unless stated otherwise, a time window width of 30 min. Each flight’s departure and arrival times are allowed to move at most 15 min earlier or later, and the maximum total change in block time is 15 min. For each flight string, the departure time of the first flight and the arrival time of the last flight are not allowed to move earlier and later, respectively. In particular, for every flight, we set $l_{x_i} = l_{y_i} = l_i = -15$ and $u_{x_i} = u_{y_i} = u_i = 15$, except for the first and last flights of each string where we set $l_{x_i} = 0$ and $u_{y_i} = 0$. Additionally, minimum passenger connection times are assumed to be 30 min.

The performances of the *minTAD* and *maxEffPaxSlack15* solutions over the period of March 1–25, 2008 are summarized in Table 2.

Schedule statistics

Recall that the robust flight schedule adjustment model allows ground time slack to be transformed into block time slack. In the *minTAD* solution, the total aircraft connection slack is decreased from the *Original* schedule by almost 40 %.

Table 2 Average performance evaluation statistics over 25 days (March 1–25, 2008)

	Original	minTAD	maxEffPaxSlack15
Schedule statistics			
Total A/C connection slack (min)	6,676.76	4,122.60	6,654.80
Total absolute block time change (min)	–	2,627.12	2,929.56
Average block time change (min)	–	10.55	0.09
Flight delay statistics			
Total propagated delay (min)	1,009.60	827.76	1,174.84
% of Flights with PD > 0	17.74	14.18	20.79
Total arrival delay (min)	3141.16	1,873.48	4,068.60
15-Min on-time performance (%)	76.53	87.49	67.07
60-Min on-time performance (%)	96.89	97.69	96.35
Passenger delay statistics			
Total pax delay (min)	260,565	178,004	313,990
Total disrupted pax (pax)	47.56	62.12	28.52
Total spilled pax (pax)	26.28	33.48	17.44

To quantify the difference between the `Original` schedule and solutions to the model, we report “Total Absolute Block Time Change”. Additionally, we also report “Average Block Time Change” to indicate overall direction of change in block times. Consistently with the decrease in aircraft connection slack, flight block times in the `minTAD` solution increase about 10 min, on average. This reflects the emerging practice of schedule padding to handle unpredictability in the U.S. airspace system (Mccartney 2010). As we will discuss shortly, the increase in block times is, in fact, the key to significant reduction in total arrival delay. From a passenger’s perspective, the resulting schedule provides more reliability, that is, with a high probability, he or she would not be late for an important meeting nor miss a connection. From an airline’s perspective, however, these additional block times can be expensive. We will elaborate on this in “`Cost`”.

In the `maxEffPaxSlack15` solution, although the total block time change is larger than in the `minTAD` solution, the total amount of aircraft connection slack remains almost the same as in the `Original` schedule, and the average block time change suggests that the changes in block times occur equally in both directions.

Flight delay analysis

To begin with, the `maxEffPaxSlack15` solution performs worse than the `Original` schedule in every flight delay statistic simply because the objective of maximizing the total expected effective passenger connection slack has no direct link to flight delay improvements.

Remarkably, total arrival delay is reduced by more than 40 % in the `minTAD` solution. The other flight delay statistics are also improved significantly, especially the 15-min on-time performance metric. In fact, the simulation results show that many flights now arrive earlier than scheduled. Table 3 summarizes the total arrival delay distribution for the `minTAD` solution. Total arrival delays are significantly reduced in every positive range.

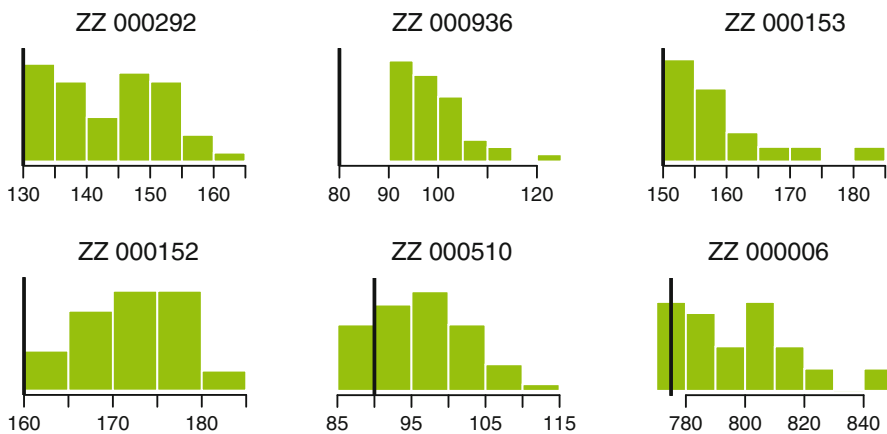


Fig. 4 Actual block time distributions

Table 3 Distributions of total arrival delays

Total arrival delay (min)	0	(0,15]	(15,60]	(60,120]	>120
Original (%)	44.33	32.20	20.36	2.17	0.94
minTAD (%)	69.90	17.59	10.20	1.59	0.73

In Chiraphadhanakul (2010), the optimization models and results for other slack re-allocation schemes are also presented and show that the aircraft re-routing and flight schedule re-timing (i.e., the block times are fixed) solutions do not achieve this large total arrival delay reduction. More specifically, the total arrival delays in the aircraft re-routing and flight schedule re-timing solutions are more than 1,000 min larger than in the minTAD solution.

According to the performance of the Original schedule, propagated delays contribute only about one-third of the total arrival delay, and another two thirds is due to independent arrival delays of each flight. Moreover, the historical data indicate that planned block times are underestimated, on average, by almost 10 min. Figure 4 depicts the actual block time distributions of some problematic flights. A vertical bar in each plot denotes the planned block time for that flight. Consequently, even though these flights experience no propagated delays, arrival delays are inevitable.

Note that in the aircraft re-routing and the flight schedule re-timing models, because block times, and thus independent arrival delays are fixed, total arrival delay minimization can only be achieved by minimizing total propagated delay. On the other hand, the block time adjustment model allows aircraft connection slack to be converted into block time slack, and hence can potentially reduce independent arrival delays in addition to propagated delays. Because the major contribution to the total arrival delay, for this particular airline, is due to independent arrival delays, the minTAD solution can effectively use additional block time slack, converted from original ground time slack, to absorb independent arrival delays and achieve the minimal total arrival delay.

Passenger delay analysis

As a result of the significant total arrival delay reduction, the total passenger delay is considerably decreased in the minTAD solution. The number of disrupted passengers, however, increases by about 30 %. Intuitively, because block times for each flight in the minTAD solution are increased by 10 min on average, most of the passenger connection times become shorter. Consequently, connecting passengers are more likely to miss their connections.

In contrast, the number of disrupted passengers is reduced by 40 % in the maxEffPaxSlack15 solution. This indicates that our objective function of maximizing the total expected effective passenger connection slack serves as a good proxy for minimizing passenger misconnections. Despite the large reduction of disrupted passengers, the maxEffPaxSlack15 solution yields higher total passenger delay than the Original schedule. Two questions arise from these results, namely

1. Why does total passenger delay decrease in the `minTAD` solution, in spite of the significant increase in disrupted passengers?
2. And why does total passenger delay increase in the `maxEffPaxSlack15` solution, in spite of the significant reduction in disrupted passengers?

These happen because total passenger delay of this particular airline is mainly driven by flight delays, not passenger misconnections. According to the passenger booking data, almost 90 % of passengers are local passengers—those that travel on a single flight leg. The passenger delay associated with these local passengers depends only on flight delays if their flights are not canceled. Moreover, we find that, on average, total *disrupted* passenger delay contributes less than 10 % of total passenger delay.

These findings suggest that for this particular airline, it is more appropriate to focus on minimizing flight delays rather than minimizing passenger misconnections. For other airlines with larger proportions of connecting passengers, however, the objective functions that minimize passenger misconnections could be more appropriate, given that the contribution of disrupted passenger delay to total passenger delay is more significant.

In summary, airline characteristics, such as a proportion of connecting passengers and contribution of disrupted passenger delay to total passenger delay, are crucial in determining which objective function is more appropriate in achieving the airline's goal to minimize total passenger delay. The objective function that works well for one airline might lead to a bad solution for other airlines.

Discussions

Alternative objectives

Table 4 summarizes the performance of the `minPD` and `maxEffACSlack15` solutions. Total propagated delay is significantly reduced in both solutions. The larger improvement in the `maxEffACSlack15` solution confirms that maximizing total expected effective aircraft slack is a better proxy for minimizing total propagated delay. Both solutions, however, result in considerably higher total arrival delay, which consequently leads to much larger total passenger delay, compared with the `minTAD` solution. Additionally, having large total arrival delays while most flights can depart on time also causes more disrupted passengers. In fact, for this model, neither minimizing the total expected propagated delay nor maximizing total expected effective aircraft connection slack is a good proxy for minimizing the total expected arrival delay.

In the aircraft re-routing and the flight schedule re-timing problems, because independent arrival delays are fixed, decreasing propagated delays will result in smaller total arrival delays, as shown in Lan et al (2006); AhmadBeygi et al (2010). This, however, is not the case for this particular model, which allows block time adjustment. One can possibly decrease propagated delays and simultaneously increase independent arrival delays, as well as total arrival delays. For instance, consider a flight that experiences a nonzero propagated delay from the preceding

flight. We can avoid this delay propagation by moving the flight's departure time to when the aircraft is ready. Without changing the flight's arrival time, the total arrival delay will remain the same in spite of the reduction in propagated delay.

We can argue similarly that maximizing the total expected effective aircraft connection slack is also not a good proxy for minimizing the total expected arrival delay in this model.

Now focus on the schedule statistics in Table 4. Block time of each flight is decreased, on average, by 6.7 and 1.4 min in the `minPD` and `maxEffACSlack15` solutions, respectively, and the total aircraft connection slack increases in both solutions. These results are opposite to the `minTAD` solution for which most of the aircraft connection slack is transformed into block time slack. This difference reflects two common approaches to building robustness into airline schedules: (1) schedule padding (i.e., increasing block time slack); and (2) turn-around time padding (i.e., increasing aircraft connection slack).

As discussed in “[Slack re-allocation approaches](#)”, block time slack provides greater flexibility than aircraft connection slack because it can absorb not only propagated delay from prior flights, but also independent departure and arrival delay (such as delays due to Ground Delay Programs, taxi delays, and airborne delays). However, it is generally more costly to add slack into block times (in particular because crew productivity is reduced and hence, crew costs are increased), and schedule padding may not be an appropriate approach for every airline. For example, an airline that operates a hub-and-spoke network may prefer turn-around time padding as the additional aircraft connection time will also facilitate passenger connections at the hubs. Interested readers are referred to Zhu (2009). In her work, she provides a comprehensive comparison of the performances of two airlines adopting these two different slack allocation approaches.

Table 4 Average performance evaluation statistics over 25 days (March 1–25, 2008) for the models with alternative objectives

	<code>minTAD</code>	<code>minPD</code>	<code>maxEffACSlack15</code>
Schedule statistics			
Total A/C connection slack (min)	4,122.60	8,303.40	7,025.64
Total absolute block time change (min)	2,627.12	2,774.88	1,876.00
Average block time change (min)	10.55	−6.72	−1.44
Flight delay statistics			
Total propagated delay (min)	827.76	606.40	588.80
% of Flights with PD > 0	14.18	8.89	8.17%
Total arrival delay (min)	1,873.48	4,705.08	3,530.04
15-Min on-time performance (%)	87.49	63.31	74.68
60-Min on-time performance (%)	97.69	96.31	96.73
Passenger delay statistics			
Total pax delay (min)	178,004	365,306	294,151
Total disrupted pax (pax)	62.12	71.20	73.96
Total spilled pax (pax)	33.48	33.16	34.16

Solution quality

We exhibit the quality of the `minTAD` solution by comparing its performance with the other two solutions, denoted by the suffixes `_expected` and `_perfectInfo`. The only difference among these solutions is the approach of using historical data. Recall that so far we consider each day of operation in January and February represents one instance of delay scenario ω and that each disruption scenario is equally likely.

For the `_expected` solution, the set of delay scenarios Ω has a single element representing the *average* independent delays of every flight, obtained from January and February data. This simple approach can be useful when only estimates of average flight delays are available. It, however, ignores the stochastic nature of delays and the correlations of delays among different flights, which can partially be captured by using many different delay scenarios. Furthermore, it can be shown that total arrival delay of each flight is a convex function of independent arrival delays of prior flights along the same aircraft route. By Jensen's inequality, for a convex function, the expected value of functions is no smaller than the function of the expected value. Therefore, it follows that the `_expected` solution underestimates the total expected arrival delay, as compared with the `minTAD` solutions. Intuitively, it is possible that an *average* independent arrival delay of a flight is so small that it does not propagate in the `_expected` solution, and consequently the total arrival delay of the subsequent flights are underestimated, although for some instances of delay scenarios the delay could be very large and potentially propagate to subsequent flights.

For the `_perfectInfo` solution, the set of delay scenarios Ω has a single element representing the *actual* independent delays of every flight. Note that this is the same set of independent delays we use to evaluate the performance of schedules (i.e., delays from March data). In other words, we solve the model as if we have perfect information about future operations. This solution provides a bound on the improvement we can possibly achieve through a particular model.

Table 5 summarizes the performance of each solution over the period of March 1–25, 2008. The `minTAD` solution performs reasonably better than the `minTAD_expected` solution with respect to every performance evaluation metric. This demonstrates the benefits of using many different delay scenarios from historical data to capture the stochasticity of delays. Nevertheless, the `minTAD_expected` solution, obtained by simply using *average* independent arrival delays of each flight, still results in a 32 % reduction in the total arrival delay and a 24 % reduction in the total passenger delay.

The performance of the `minTAD_perfectInfo` solution indicates that, given perfect information, total arrival delay can be reduced by almost 50 %, and total passenger delay can be reduced by almost 40 %. The performance gap between the `minTAD` and `minTAD_perfectInfo` solutions with respect to total arrival delay and total passenger delay is about 10 %.

Allowable changes in flight schedule

Flight schedule adjustments may result in changes in planning of other resources and demands for flights. Therefore, it might be preferable to keep the adjustments

Table 5 Average performance evaluation statistics over 25 days (March 1–25, 2008) of the minTAD solutions with different approaches of using historical data

	minTAD	minTAD _expected	minTAD _perfectInfo
Schedule statistics			
Total A/C connection slack (min)	4,122.60	4,756.52	5,175.28
Total absolute block time change (min)	2,627.12	2,039.52	1,753.72
Average block time change (min)	10.55	7.93	6.20
Flight delay statistics			
Total propagated delay (min)	827.76	911.84	629.00
% of Flights with PD > 0 (%)	14.18	17.90	9.14
Total arrival delay (min)	1,873.48	2,112.56	1,631.12
15-Min on-time performance (%)	87.49	85.12	89.83
60-Min on-time performance (%)	97.69	97.42	97.97
Passenger delay statistics			
Total pax delay (min)	178,004	196,815	159,013
Total disrupted pax (pax)	62.12	70.12	52.64
Total spilled pax (pax)	33.48	36.48	28.24

minimal. To demonstrate the effect of time window width on the performance of solutions, we solve for the minTAD solutions again with time window widths of 20 and 10 min. Table 6 summarizes the performance of the solutions.

As the time window decreases, the flexibility of the model is limited, and thus the total arrival delay reduction decreases from 40 % in the 30-min time window case

Table 6 Average performance evaluation statistics over 25 days (March 1–25, 2008) with different time window widths

Time window	±15	±10	±5
Schedule statistics			
Total A/C connection slack (min)	4,122.60	4,747.64	5,616.48
Total absolute block time change (min)	2,627.12	1,972.08	1,079.72
Average block time change (min)	10.55	7.97	4.38
Flight delay statistics			
Total propagated delay (min)	827.76	874.48	939.24
% of Flights with PD > 0 (%)	14.18	14.78	16.07
Total arrival delay (min)	1,873.48	2,100.88	2,513.08
15-Min on-time performance (%)	87.49	85.69	82.51
60-Min on-time performance (%)	97.69	97.52	97.27
Passenger delay statistics			
Total pax delay (min)	178,004	192,788	219,148
Total disrupted pax (pax)	62.12	55.24	49.88
Total spilled pax (pax)	33.48	31.12	27.40

to 33 and 22 % in the 20- and 10-min cases, respectively. These remarkable improvements suggest that by increasing block times by less than 5 min on average, an airline can significantly improve its schedule performance.

Cost

Robustness in an airline schedule is typically achieved at a cost. As discussed in “[The airline schedule planning process](#)”, flight schedule planning is the first step in the airline sequential planning process. Therefore, flight schedule adjustments resulting from the optimization model may cause changes in the planning of other resources—potentially with higher cost. The current model already ensures that every aircraft and passenger connection in the original schedule remains feasible. However, due to lack of information, we ignore the potential effects of our model on crew schedules.

With the multi-faceted work rules such as minimum rest time, maximum flying time, and maximum duty period, crew duties tend to be affected when a flight schedule is adjusted. Because crews are paid at least for the scheduled block time, block time slack is generally considered more costly than ground time slack. Additionally, longer block times can result in reductions in the number of flights that one crew can operate, or reductions in the number of possible crew connections, all resulting in reduced crew productivity. Consequently, the planned costs associated with the new schedule may increase, even though time windows within which a flight schedule is allowed to be adjusted are small.

On the other hand, a robust schedule leads to fewer delays and disruptions and hence smaller recovery costs. We believe that airline savings from delay reduction could outweigh the potential increase in planned costs and result in smaller *realized* costs, compared with those of non-robust schedules. Because different airlines have different cost structures and experience different levels of disruptions in their operations, the benefits of robust schedules obtained from the model can vary, however.

Conclusion

This paper provides a mix of modeling, methodology, and the provision of operational and managerial insights. We formulate a flight schedule adjustment model that strategically re-allocates existing schedule slack to minimize some proxy of expected delays or disruptions. Alternative objective functions that can potentially result in more robust solutions with respect to different performance metrics are also presented. Importantly, we introduce a novel notion of *effective slack*, which is proved to serve as a good robustness proxy.

Given historical data from an international carrier, we use a scenario-based approach to obtain the proof-of-concept results and evaluate schedule performance. Unlike many papers in the literature that evaluate the resulting schedules based solely on the objective function values, we evaluate the performance of the resulting schedules on multiple performance metrics to understand the impacts of different

objective functions. Additionally, different datasets are used for optimization and evaluation to reflect the actual planning process, in which the observed uncertainty might not fully represent future events.

The results show that minor modifications to an original schedule can significantly improve the overall performance of the schedule, even in the case where the changes are limited. Through empirical experiments, we also address an open question in the literature concerning how one slack re-allocation approach is more advantageous than another for a given airline with certain characteristics. Some of our findings are summarized as follows:

For the airline we study, the major contribution to total arrival delay is from independent arrival delays (such as delays caused by Ground Delay Programs, taxi delays, and airborne delays), rather than propagated delays from preceding flights. Because only block time slack can absorb such delays, using models that are incapable of altering block time (e.g., aircraft re-routing and flight schedule re-timing models) does not lead to significant arrival delay reductions.

Given the complexity of the passenger delay calculation, we cannot model total passenger delay as an objective function in our formulation. Two possible proxies for minimizing total passenger delay are (1) to minimize a particular flight delay metric or (2) to minimize passenger disruptions. The percentage of connecting passengers and the percentage of total passenger delay due to disrupted passengers are two key statistics that can be used to determine which proxy is more appropriate. For this particular airline, almost 90 % of passengers travel on a single flight leg, and total *disrupted* passenger delay represents less than 10 % of total passenger delay. Therefore, minimizing flight delay serves as a better proxy for minimizing passenger delay.

For the proposed model, minimizing total arrival delay and minimizing total propagated delay lead to two different approaches to building robustness into airline schedules: (1) schedule padding, and (2) turn-around time padding. Although both are common in practice, they affect the performance and operating costs of an airline differently, and therefore, airlines may prefer different approaches depending on their operations.

Acknowledgements The authors are grateful for financial support of this project by Jeppesen, and they thank the anonymous reviewers for their useful comments.

References

- Ageeva Y (2000) Approaches to incorporating robustness into airline scheduling. Master's thesis, Massachusetts Institute of Technology
- AhmadBeygi S, Cohn A, Lapp M (2010) Decreasing airline delay propagation by re-allocating scheduled slack. *IIE Trans* 42(7):478–489
- Ball M, Barnhart C, Dresner M, Hansen M, Neels K, Odoni A, Peterson E, Sherry L, Trani A, Zou B (2010) Total delay impact study: a comprehensive assessment of the costs and impacts of flight delay in the united states. Tech rep
- Belobaba P, Odoni A, Barnhart C (2009) *The global airline industry*. Wiley, New York
- Ben-Tal A, Nemirovski A (1998) Robust convex optimization. *Math Oper Res* 23(4):769–805
- Bertsimas D, Sim M (2004) The price of robustness. *Oper Res* 52(1):35–53
- Birge J, Louveaux F (1997) *Introduction to stochastic programming*. Springer, Berlin

- Borndörfer R, Dovica I, Nowak I, Schickinger T (2010) Robust tail assignment. Tech Rep 10-08, ZIB, Takustr.7, 14195 Berlin
- Bratu S (2003) Airline passenger on-time schedule reliability: analysis, algorithms and optimization decision models. Ph.D. thesis, Massachusetts Institute of Technology
- Bratu S, Barnhart C (2005) An analysis of passenger delays using flight operations and passenger booking data. *Air Traffic Control Q* 13(1):1–28
- Burke EK, Causmaecker PD, Maere GD, Mulder J, Paelinck M, Berghe GV (2009) A multi-objective approach for robust airline scheduling. *Comput Oper Res*
- Cadarso L, Á Marín (2013) Robust passenger oriented timetable and fleet assignment integration in airline planning. *J Air Transp Manag* 26(0):44–49
- Chiraphadhanakul V (2010) Routing and scheduling models for robust allocation of slack. Master's thesis, Massachusetts Institute of Technology
- Dunbar M, Froyland G, Wu CL (2012) Robust airline schedule planning: minimizing propagated delay in an integrated routing and crewing framework. *Transport Sci* 46(2):204–216
- Eggenberg N (2009) Combining robustness and recovery for airline schedules. Ph.D. thesis, Ecole Polytechnique Federale de Lausanne
- Eggenberg N, Salani M, Bierlaire M (2011) Uncertainty feature optimization: an implicit paradigm for problems with noisy data. *Networks* 57(3):270–284
- Ehrgott M, Ryan D (2000) Bicriteria robustness versus cost optimisation in tour of duty planning at air new zealand. In: Proceedings of the 35th annual conference of the operational research society of New Zealand, pp 31–40
- Fearing D, Vaze V, Barnhart C (2010) Modeling passenger travel and delays in the national air transportation system. *Oper Res* (Submitted)
- Gao C, Johnson E, Smith B (2009) Integrated airline fleet and crew robust planning. *Transport Sci* 43(1):2–16
- Kang L (2004) Degradable airline scheduling: an approach to improve operational robustness and differentiate service quality. PhD thesis, Massachusetts Institute of Technology
- Kleywegt A, Shapiro A, Homem-de Mello T (2002) The sample average approximation method for stochastic discrete optimization. *SIAM J Optim* 12(2):479–502
- Lan S (2003) Planning for robust airline operations: optimizing aircraft routings and flight departure times to achieve minimum passenger disruptions. PhD thesis, Massachusetts Institute of Technology
- Lan S, Clarke J, Barnhart C (2006) Planning for robust airline operations: optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transport Sci* 40(1):15–28
- Lapp M, Cohn A (2012) Modifying lines-of-flight in the planning process for improved maintenance robustness. *Comput Oper Res* 39(9):2051–2062
- Marla L, Barnhart C (2010) Robust optimization: lessons learned from aircraft routing, working paper
- Mccartney S (2010) Why a 6-h flight now takes seven. *Wall Street J* (4 Feb 2010). <http://online.wsj.com/article/SB10001424052748703575004575043143789789222.html>
- Homem-de Mello T (2000) Monte carlo methods for discrete stochastic optimization. In: Uryasev S, Pardalos PM (eds) *Stochastic optimization: algorithms and applications*, vol 54. Kluwer Academic Publishers, Dordrecht, pp 95–117
- Rosenberger J, Johnson E, Nemhauser G (2004) A robust Fleet-Assignment model with hub isolation and short cycles. *Transport Sci* 38(3):357–368
- Ruszczynski A, Shapiro A (eds) (2003) *Stochastic programming, handbooks in operations research and management science*, vol 10. Elsevier, Amsterdam
- Schaefer A, Johnson E, Kleywegt A, Nemhauser G (2005) Airline crew scheduling under uncertainty. *Transport Sci* 39(3):340–348
- Shapiro A (2003) Monte carlo sampling methods. In: Ruszczynski A, Shapiro A (eds) *Stochastic programming, handbooks in operations research and management science*, vol 10. Elsevier, Amsterdam pp 353–425
- Shebalov S, Klabjan D (2006) Robust airline crew pairing: move-up crews. *Transport Sci* 40(3):300–312
- Smith B, Johnson E (2006) Robust airline fleet assignment: imposing station purity using station decomposition. *Transport Sci* 40(4):497–516
- Sohoni M, Lee YC, Klabjan D (2011) Robust airline scheduling under block-time uncertainty. *Transport Sci* 45:451–464
- Soyster AL (1973) Technical note: convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper Res* 21(5):1154–1157

- Uryasev S, Pardalos PM (eds) (2000) Stochastic optimization: algorithms and applications, vol 54. Kluwer Academic Publishers, Dordrecht
- Yen J, Birge J (2006) A stochastic programming approach to the airline crew scheduling problem. *Transport Sci* 40(1):3–14
- Zhu Y (2009) Evaluating airline delays: the role of airline networks, schedules, and passenger demands. Master's thesis, Massachusetts Institute of Technology