Transportation Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Robust Aircraft Routing

Chiwei Yan, Jerry Kung

TRANSPORTATION SCIENCE
Articles in Advance, pp. 1–17
ISSN 0041-1655 (print) | ISSN 1526-5447 (online)

http://dx.doi.org/10.1287/trsc.2015.0657

Robust Aircraft Routing

Chiwei Yan, Jerry Kung
Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
{chiwei@mit.edu, jkung@mit.edu}

We propose a robust optimization approach to minimize total propagated delay in the aircraft routing problem. Instead of minimizing the expected total propagated delay by assuming that flight leg delays follow specific probability distributions, our model minimizes the maximal possible total propagated delay when flight leg delays lie in a prespecified uncertainty set. We develop exact and tractable solution approaches for our robust model. The major contribution of our model is that it allows us to explicitly model and handle correlation in flight leg delays (e.g., because of weather or various air traffic management initiatives) that existing approaches cannot efficiently incorporate. Using both historical delay data and simulated data, we evaluate the performance of our model and benchmark against the state-of-the-research stochastic approach. In most of the cases, we observe that our model outperforms the existing approach in lowering the mean, reducing volatility, and mitigating extreme values of total propagated delay. In the cases where a deficit in one of the three criteria exists, gains in the other two criteria usually offset this disadvantage. These results suggest that robust optimization approaches can provide promising results for the aircraft routing problem.

Keywords: aircraft routing; robust airline planning; delay propagation; robust optimization
History: Received: November 2014; revision received: July 2015; accepted: August 2015. Published online in *Articles in Advance* July 5, 2016.

1. Introduction

Airline delays are prevalent and costly. In 2013, almost one out of every four flight legs operated by major U.S. carriers arrived more than 15 minutes late; of these delayed flights, one-third were a result of propagated delay: the late arrival of an aircraft causing a late departure (Bureau of Transportation Statistics 2013). One of the key drivers for these propagated delays is that in the past airlines have employed optimization models in an attempt to maximize profit; this has often led to the creation of tight schedules in an attempt to increase aircraft utilization and decrease crew salaries for time on the ground (Klabjan et al. 2001). As a consequence, delays propagate very rapidly throughout the network. This in turn leads to significant costs for an airline, such as additional pay for the crew, customer dissatisfaction, and further decreased utilization of aircraft because of flight cancellations.

To mitigate the discrepancy between what is planned and what actually happens, the research community has developed robust planning methods to proactively consider such delays and disruptions. These methods create schedules with the objective of constructing plans that have one of two kinds of robustness built in: (1) ease of repair once disrupted or (2) diminished propensity of delay disturbances.

Within the first category, Kang (2004) partitions flight schedules into independent layers so that delay or disruption in one layer will not affect the others. Each layer is assigned priority based on its expected revenue. Flights in higher layers are protected most, whereas those in lower layers are more likely to experience delay or cancellation. This approach significantly reduces delay and disruption costs. Rosenberger, Johnson, and Nemhauser (2004) develop a string-based fleet assignment model that embeds many short cancellation cycles and limits the number of aircraft that can serve each hub. Such characteristics prevent the cancellation of a series of flights and isolate the disruption to a particular hub. Smith and Johnson (2006) introduce the concept of "station purity," which limits the number of fleet types that can serve each airport. This creates additional aircraft swapping opportunities. Shebalov and Klabjan (2006) propose a robust crew pairing model that maximizes the number of crews that can potentially be swapped in operations. Yen and Birge (2006) develop a two-stage stochastic integer program with recourse for the crew scheduling problem. Their model produces a robust crew schedule by incorporating disruptions in the evaluation of crew schedules during the long-range planning phase. Recently, Froyland, Maher, and Wu (2014) present a recoverable robust aircraft routing model as a two-stage stochastic program. They explicitly model the aircraft routing decision in the first stage, as well as a full set of recovery operations including flight cancellation, delay, and aircraft swapping in the second-stage recourse problem.

Our line of research falls into the second category of built-in robustness, where we aim to devise an airline schedule that is less susceptible to delay disturbance. One existing idea in this line of work has been to reduce delay propagation by retiming the flight schedule. Ahmadbeygi, Cohn, and Lapp (2010) use a propagation tree to minimize propagated delay so that slack in the schedule can be reallocated to where it is required most. In this research, we do not study the retiming idea. Instead, the focus of this paper is on building robustness into the aircraft routing problem to protect against propagated delay. Aircraft routing is a critical airline planning phase in which the goal is to create a sequence of flight legs to be operated by individual aircraft such that each flight is covered in exactly one routing, and all aircraft are properly maintained. Typical maintenance requirements include, for example, aircraft inspection for every 100 hours of flight time at certain maintenance-compatible stations (Federal Aviation Administration 2014), and so on. The aircraft routing problem is solved separately for each fleet type. As Lan, Clarke, and Barnhart (2006) indicate, this problem can usually be cast as a feasibility problem, thus providing flexibility to achieve desired robustness by designing an appropriate objective function. To illustrate the impact of aircraft routing on delay propagation, suppose two connecting fights are close to each other with respect to timing (the arrival time of the first flight is very close to the departure time of the second flight). In this case, it is not desirable to fly both using a single aircraft, because even a small amount of delay from the first flight will end up propagating to the second flight.

The idea of reducing propagated delay through clever routing is not new. Some of the pioneers include Lan, Clarke, and Barnhart (2006), who develop a stochastic integer program to minimize the total expected propagated delay along aircraft routes where all of the flight leg delays are fitted as independent log-normal distributions using historical data. To solve the program, they design an approximation algorithm based on column generation. To our knowledge, no efficient solution approach that can solve this model to optimality exists, even when flight leg delays are assumed to be independent random variables. This is perhaps due to the nonlinearity in the computation of propagated delay. Weide, Ryan, and Ehrgott (2010) propose an iterative approach for a robust and integrated aircraft routing and crew pairing problem to mitigate delay propagation. Dunbar, Froyland, and Wu (2012) propose an improved solution approach to solve a deterministic integrated aircraft routing and crew pairing problem to minimize total propagated delay, given that flight leg delays are known to be constant. This work was recently extended to incorporate stochastic flight leg delay information (Dunbar, Froyland, and Wu 2014). They use historical flight leg delay data to construct random scenarios and develop two algorithms to solve the stochastic program. Borndörfer et al. (2010) develop an alternative robust aircraft routing formulation whose objective is to minimize the sum of the probability when propagated delay is positive over all flights, assuming that flight leg delays follow certain independent probability distributions. They demonstrate the effectiveness of their model in reducing the amount of propagated delay through computational experiments, but in theory it is unclear how this objective function is related to the actual amount of propagated delay.

In this work, we depart from previous methods (Lan, Clarke, and Barnhart 2006; Dunbar, Froyland, and Wu 2014) that use *stochastic optimization* to minimize the expectation of propagated delay. We instead utilize *robust optimization* to minimize the maximal possible propagated delay. By contrast to the previous literature, which models flight leg delays as probability distributions, we utilize uncertainty sets to capture the stochasticity of flight leg delays. The major motivation of applying robust optimization to the aircraft routing problem comes from the ability to handle correlation in flight leg delays. Flights arriving to the same airport at similar times are usually delayed because of various air traffic management initiatives or inclement weather. Thus, incorporating delay correlation into robust aircraft planning could potentially yield additional benefits. As mentioned earlier, no efficient methods capable of dealing with delay correlation data have been explored in the robust aircraft routing literature.

On the other hand, robust optimization has been successfully applied to a number of application areas where the problems are combinatorially hard and involve correlated and complex uncertainty (Bertsimas, Brown, and Caramanis 2011). Although a major criticism of the robust optimization-based approach is the conservativeness of optimizing against the worst case as opposed to the expectation, under many circumstances, it may still perform better than stochastic optimization in expectation because of proper utilization of correlation data. This is mainly due to the added tractability that robust optimization enjoys, which allows it to handle complex uncertainty. Furthermore, additional protection is built into robust optimization, especially when realized data differs significantly from historical data (Bertsimas and Thiele 2006; Bertsimas et al. 2013b).

Applying robust optimization to the aircraft routing problem is fairly new. Until now, only one study (Marla and Barnhart 2010) has considered this approach. They compare the performance of three different classes of models: (1) chance-constrained programming, (2) robust optimization, and (3) stochastic optimization (Lan, Clarke, and Barnhart 2006). In their consideration of robust optimization, they model flight leg delays using a budget uncertainty set developed by Bertsimas and Sim (2004). Although their conclusions do not favor the robust optimization approach, we believe that by modeling uncertainty sets differently and solving the resultant robust problem cleverly, we may still be able to improve on existing methods. Furthermore, we aim to provide comprehensive benchmarks to better evaluate the value of this approach.

The remainder of this paper is organized as follows. In §2, we present our robust formulation of the robust aircraft routing problem in detail. We also discuss how to model the uncertainty set that captures flight leg delays. Our uncertainty set is a polyhedral set based on the central limit theorem and incorporates correlation structure in flight leg delays. In §3, we develop an exact decomposition solution approach for our robust model based on column-and-row generation. We also discuss detailed computational approaches to solving the resulting separation and pricing subproblems. In §4, we conduct extensive computational experiments to evaluate the benefits of our robust model and benchmark it against a state-of-the-research model. We conclude with a discussion in §5.

2. The Robust Aircraft Routing Problem Formulation

In this section, we describe our formulation for the robust aircraft routing problem using the framework of robust optimization. We first outline our mathematical formulation and then discuss how we model and construct uncertainty sets for flight leg delays.

2.1. Robust Aircraft Routing Formulation

The aircraft routing problem can be formally stated as follows: given a set of aircraft of a specific fleet type and a set of flights that must be operated, determine how the fleet can be routed so that each flight leg is included in exactly one aircraft routing, and all aircraft are properly maintained. Among all feasible assignments of aircraft to routes, we seek the one that incurs the least amount of total propagated delay.

We denote by $\mathcal{R}$ the set of all feasible routes, $\mathcal{F}$ the set of flights, and N the total number of available aircraft. For each $r \in \mathcal{R}$, $F(r)$ represents the sequence of flights along route r. All maintenance-feasible routes can be represented by the columns of an $|\mathcal{F}| \times |\mathcal{R}|$ binary matrix A, where $A_{i,j} = 1$ if route j visits flight i, and $A_{i,j} = 0$ otherwise. Binary decision variables x_r denote whether or not route r is chosen in the optimal solution, with $x_r = 1$ if it is selected, and $x_r = 0$ otherwise. As in Lan, Clarke, and Barnhart (2006) and Dunbar, Froyland, and Wu (2012), we consider two kinds of delays in our model:

1. *Primary delay*, denoted d_j for each flight leg $j \in \mathcal{F}$, which includes en-route delay, passenger connection delay, ground handling delay, and other delays that are not a function of routing.
2. *Propagated delay*, denoted p_j^r for flight leg $j \in F(r)$ on route $r \in \mathcal{R}$, represents the amount of delay propagated to flight j caused by the late arrival of the upstream flight.

Given flight leg primary delay $\mathbf{d} \in R_+^{|\mathcal{F}|}$, the propagated delay $\mathbf{p}^r(\mathbf{d}) \in R_+^{|F(r)|}$, $r \in \mathcal{R}$ can be calculated iteratively as a function of $\mathbf{d}$. Suppose flights (i, j) are two consecutive flights on route r, $\text{slack}_{i,j}$ denotes the slack time for a connection (i, j), which is the difference between the scheduled arrival time of flight i and the scheduled departure time of flight j, minus the mean turnaround time for the corresponding aircraft type. The propagated delay to flight j from flight i then follows the expression

$$p_j^r(\mathbf{d}) = \max\{0, p_i^r(\mathbf{d}) + d_i - \text{slack}_{i,j}\}, \quad \forall (i, j) \in r, \ \forall r \in \mathcal{R}. \tag{1}$$

With the above notation, given flight leg delays $\mathbf{d}$, we write the deterministic aircraft routing problem (AR) with the objective of minimizing total propagated delay as the following integer program:

$$\text{(AR)} \quad \min_{\mathbf{x}} \ \sum_{r\in\mathcal{R}} c_r(\mathbf{d}) x_r = \sum_{r\in\mathcal{R}} \boxed{\sum_{i\in F(r)} p_i^r(\mathbf{d})} x_r \tag{2}$$

$$\text{s.t.} \quad A\mathbf{x} = \mathbf{e}, \tag{3}$$

$$\sum_{r\in\mathcal{R}} x_r \le N, \tag{4}$$

$$\mathbf{x} \in \{0, 1\}^{|\mathcal{R}|}.$$

Objective (2) is the sum of propagated delay over all flights, given constant flight leg primary delays $\mathbf{d}$. Constraints (3) are the flight cover constraints that ensure that each flight leg must be covered by only one aircraft routing. Constraints (4) are the fleet count constraints that keep the total number of aircraft used to be less than or equal to the number of available aircraft. To enhance the robustness of this model against all flight leg primary delays lying in a prespecified uncertainty set $\mathcal{U}$, we consider the following robust aircraft routing (RAR) formulation:

$$\text{(RAR)} \quad \min_{\mathbf{x}} \max_{\mathbf{d}\in\mathcal{U}} \ \sum_{r\in\mathcal{R}} c_r(\mathbf{d}) x_r = \sum_{r\in\mathcal{R}} \sum_{i\in F(r)} p_i^r(\mathbf{d}) x_r \tag{5}$$

$$\text{s.t.} \quad A\mathbf{x} = \mathbf{e}, \tag{6}$$

$$\sum_{r\in\mathcal{R}} x_r \le N, \tag{7}$$

$$\mathbf{x} \in \{0, 1\}^{|\mathcal{R}|}.$$

Objective (5) is to minimize the maximum possible total propagated delay when individual primary flight delays $\mathbf{d}$ are drawn from the uncertainty set $\mathcal{U}$.

2.2. Modeling Uncertainty Set $\mathcal{U}$ for Flight Leg Primary Delays d

By contrast to existing literature (Lan, Clarke, and Barnhart 2006; Borndörfer et al. 2010), in which it is assumed that primary delays on flight legs $\mathbf{d}$ are independent for the purposes of tractability, we believe that flight legs have a clear dependence structure that should be accounted for to improve solution quality.

As an example, consider two different flights arriving at the same airport within the same hour. Both of these flights will be affected by the same weather conditions—one of the main drivers of primary delay—and may be delayed by similar amounts of time. Furthermore, various air traffic management initiatives are designed to delay flights that share the same characteristics. For instance, ground delay programs (GDPs) assign delays to flights entering into the same destination airport, and airspace flow programs (AFPs) control flights flying through the same flow constrained area (FCA). Moreover, different flow management programs often have complex interactions with each other (Barnhart et al. 2012). Thus, correlation of primary delay between different flights is a natural phenomenon that should be considered when generating aircraft routes.

With this insight in mind, we allow covariance data to play a central role in our uncertainty sets. Using historical schedule data from the Airline Service Quality Performance (ASQP) database, we first compute the primary delay for each flight leg by subtracting the propagated delay from total arrival delay based on the algorithm provided in Lan, Clarke, and Barnhart (2006). We then calculate the sample means $\hat{\mu} \in R_{+}^{|\mathcal{F}|}$ of primary delays for each flight leg and create a sample covariance matrix $\hat{\Sigma} \succeq 0$, $\hat{\Sigma} \in R^{|\mathcal{F}| \times |\mathcal{F}|}$. We then create the following uncertainty sets for flight leg delays $\mathbf{d}$, where $C = \hat{\Sigma}^{-1/2}$, and $\hat{\sigma}_f$ denotes the standard deviation of the primary delay for flight $f \in \mathcal{F}$:

$$\mathcal{U}_p := \Big\{ \mathbf{d} \in \mathbb{R}_{+}^{|\mathcal{F}|} \mid \|\mathbf{C}(\mathbf{d} - \hat{\mu})\|_p \le \sqrt{|\mathcal{F}|} \cdot \Gamma; \quad \left|(d_f - \hat{\mu}_f)/\hat{\sigma}_f\right| \le \Gamma, \ \forall f \in \mathcal{F} \Big\}, \tag{8}$$

where $p \in [0, +\infty]$, $\Gamma \in [0, +\infty)$ are two exogenous parameters. The first constraint in the uncertainty set is inspired by the central limit theorem and is commonly used in the robust optimization literature. To see this, when primary leg delays $\mathbf{d}$ are uncorrelated, $C = \mathrm{diag}(1/\sigma_f)_{f \in \mathcal{F}}$, the first constraint is reduced to $\|\mathrm{diag}(1/\sigma_f)_{f \in \mathcal{F}} (\mathbf{d} - \mu)\|_p \le \sqrt{|\mathcal{F}|} \cdot \Gamma$. In the case $p = 2$, this uncertainty set is an ellipsoid centered around μ, where the length of each semiprincipal axis is the value of the standard deviation of primary delays for the corresponding particular flight. When the primary delays are correlated, the uncertainty set would correspond to a stretching and rotation of the ellipsoid. For example, when primary delays for two flights are positively correlated, the uncertainty set would capture simultaneous large and small delays for pairs of correlated flights and would lose some regions where one flight delay is large, and the other is small, and vice versa. We also add box constraints for each individual primary delay d_f to control the degree to which each individual primary delay can vary from its historical mean. The parameter Γ, commonly referred to as the budget of uncertainty, controls the protection level we want: the larger the Γ, the more conservative we are in the sense that we protect against delays that are allowed to lie in a larger set. For example, for this central limit theorem set (8), $\Gamma = 3$ usually gives us a high level of protection when the data follows a normal distribution, because almost all of the probability mass lies in the range of $[\mu - 3\sigma, \mu + 3\sigma]$ for normal random variables. We refer interested readers to Bertsimas, Pachamanova, and Sim (2004) for more details regarding this uncertainty set.

We choose $p = 1$ (L_1 norm) and introduce auxiliary variables y_i, $i \in \{1, \ldots, |\mathcal{F}|\}$ so that $\mathcal{U}$ can be equivalently formulated as a polyhedral set

$$\mathcal{U} := \Big\{ \mathbf{d} \in \mathbb{R}_{+}^{|\mathcal{F}|} \mid \exists\, y \in R^{|\mathcal{F}|} \ \text{s.t.} \sum_{i=1}^{|\mathcal{F}|} y_i \le \sqrt{|\mathcal{F}|} \cdot \Gamma; \ \pm c_i^T (\mathbf{d} - \hat{\mu}) \le y_i, \ \forall\, i \in \{1, \ldots, |\mathcal{F}|\}; \ \left|(d_f - \hat{\mu}_f)/\hat{\sigma}_f\right| \le \Gamma, \ \forall f \in \mathcal{F} \Big\}, \tag{9}$$

where c_i^T is the ith row of matrix C.

3. Solution Approach

In this section, we describe our solution approach via a column-and-row generation framework for the (RAR) model introduced in §2.1. We also discuss our computational approaches for solving the separation problem and the pricing problem, respectively.

3.1. Column-and-Row Generation Framework

As in the deterministic aircraft routing problem (2)–(4), (RAR) contains a huge number of potential decision variables (aircraft routes). Because it is impractical to enumerate all feasible routes explicitly, branch and price (Barnhart et al. 1998b) is often used to solve such problems. Unfortunately, this means that the traditional dualization approach for obtaining the robust counterpart (RC) introduced by Ben-Tal and Nemirovski (1999) cannot be applied in this setting, because we do not have the full set of decision variables included in the model before starting the solution process. However, we can still apply an iterative cutting-plane method (Bertsimas, Dunning, and Lubin 2016) to repeatedly solve (RAR) with a finite subset of the constraints. At each iteration, we check whether any violated constraints can be generated, adding them if they exist, and resolving until there are no more violated constraints. To use the cutting-plane method, we consider the equivalent epigraph reformulation (epi-RAR) for the RAR model

$$\text{(epi-RAR)} \quad \min_{\mathbf{x}, z} \; z$$

$$\text{s.t.} \quad \sum_{r\in\mathcal{R}} \sum_{i\in F(r)} p_i^r(\mathbf{d}) x_r \le z, \quad \forall\, \mathbf{d}\in\mathcal{U}, \tag{10}$$

$$A\mathbf{x} = \mathbf{e}, \tag{11}$$

$$\sum_{r\in\mathcal{R}} x_r \le N, \tag{12}$$

$$\mathbf{x}\in\{0,1\}^{|\mathcal{R}|}.$$

Note that (epi-RAR) is a mixed-integer linear program with an infinite number of rows (constraint (10)) and a huge number of columns. To address this, we utilize efficient decomposition methods on both rows and columns. We refer to constraints (10) as *robustifying constraints* because they help to protect against all possible flight primary delays in the uncertainty set.

3.1.1. The Separation Problem. For the relaxed problem (epi-RAR) with only a subset of robustifying constraints (10), we let $\mathbf{x}^*$ denote the optimal solution, and we let z^* denote the optimal objective value. We denote by $\mathcal{R}_{\mathbf{x}^*}$ the feasible aircraft routes that are specified by $\mathbf{x}^*$ (i.e., $\mathcal{R}_{\mathbf{x}^*} = \{r\in\mathcal{R} \mid x_r^* = 1\}$). The following separation problem (SEP) checks whether any violated constraints in (10) must be added into the model:

(SEP)

$$z(\mathcal{R}_{\mathbf{x}^*}) = \max_{\mathbf{p},\mathbf{d}} \sum_{r\in\mathcal{R}_{\mathbf{x}^*}} \sum_{i\in F(r)} p_i^r \tag{13}$$

$$\text{s.t.} \quad p_j^r = \max\{0, p_i^r + d_i - \text{slack}_{i,j}\}, \quad \forall\, r\in\mathcal{R}_{\mathbf{x}^*},\ \forall\,(i,j)\in r, \tag{14}$$

$$\mathbf{d}\in\mathcal{U}.$$

Given a specific aircraft routing, (SEP) optimizes the flight leg primary delays $\mathbf{d}\in\mathcal{U}$ to maximize the total propagated delay stated in objective (13). We note here that (SEP) cannot be solved by separating into parallel formulations, one for each route. This is because primary delays for flights from different routes could be jointly constrained in uncertainty set $\mathcal{U}$.

Constraint (14) is nonlinear in nature, but we can linearize it through a big-M reformulation as follows, denoted as (SEP-bigM):

(SEP-bigM)

$$z(\mathcal{R}_{\mathbf{x}^*}) = \max_{\mathbf{p},\mathbf{d}} \sum_{r\in\mathcal{R}_{\mathbf{x}^*}} \sum_{i\in F(r)} p_i^r \tag{15}$$

$$\text{s.t.} \quad p_j^r \le p_i^r + d_i - \text{slack}_{i,j} + M_j^1 I_j, \quad \forall\, r\in\mathcal{R}_{\mathbf{x}^*},\ \forall\,(i,j)\in r, \tag{16}$$

$$p_j^r \le 0 + M_j^2(1-I_j), \quad \forall\, r\in\mathcal{R}_{\mathbf{x}^*},\ \forall\,(i,j)\in r, \tag{17}$$

$$\mathbf{d}\in\mathcal{U},$$

$$I\in\{0,1\}^{|\mathcal{F}|},$$

where M_j^1, M_j^2 are sufficiently large constants (big-Ms), and I are auxiliary indicator variables that are explained in more detail below.

Since $\mathcal{U}$ is a polyhedral set in our case, (SEP-bigM) is a mixed-integer linear program. Along with the objective, constraints (16) and (17) together restrict that $p_j^r = \max\{0, p_i^r + d_i - \text{slack}_{i,j}\}, (i,j)\in r$, which is the definition of propagated delay. To see this, if auxiliary variable $I_j = 1$, constraints (16) become ineffective, and constraints (17) indicate $p_j^r = 0$ because of the maximization of the objective function; similarly, if $I_j = 0$, constraints (16) ensure $p_j^r = p_i^r + d_i - \text{slack}_{i,j}$. Thus at the optimal solution, I_j will automatically pick the right value to ensure $p_j^r = \max\{0, p_i^r + d_i - \text{slack}_{i,j}\}$. We solve (SEP-bigM) with route $\mathcal{R}_{\mathbf{x}^*}$ and denote the optimal flight leg primary delay allocation by $\mathbf{d}^*$. If $z(\mathcal{R}_{\mathbf{x}^*}) \le z^*$, then no violated constraint need be added; if $z(\mathcal{R}_{\mathbf{x}^*}) > z^*$, we add the violated constraint $\sum_{r\in\mathcal{R}} \sum_{i\in F(r)} p_i^r(\mathbf{d}^*) x_r \le z$ to the relaxed master problem and resolve.

Solving (SEP-bigM) is not an easy task because of the combinatorial structure of mixed-integer programs. However, tightening the big-M constants for each constraint can make the formulation substantially stronger than if big-Ms are assigned arbitrarily large values. We present a method below to determine tightened but sufficiently large big-Ms.

Proposition 1. *Let $d_j^* = \max_{\mathbf{d}\in\mathcal{U}} d_j$, $\forall\, j\in\{1,\ldots,|\mathcal{F}|\}$, and $p_j^r(\mathbf{d}^*)$, $\forall\, j\in\{1,\ldots,|\mathcal{F}|\}$ be the corresponding propagated delays under flight leg delays $\mathbf{d}^* = \{d_1^*,\ldots,d_{|\mathcal{F}|}^*\}$. Then $M_j^1 = \text{slack}_{i,j}$, $M_j^2 = p_j^r(\mathbf{d}^*)$ are valid values of big-Ms for* (SEP-bigM).

Proof. In the optimal solution, indicator variable I_j is 1 if $p_i^r + d_i - \text{slack}_{i,j} \le 0$, which makes $p_j^r = \max\{0, p_i^r + d_i - \text{slack}_{i,j}\} = 0$. Since constraint (17) becomes $p_j^r \le 0$ under $I_j = 1$, to let $p_j^r = 0$ in the optimal solution, the right-hand side of constraint (16) should be greater than or equal to 0, i.e., $p_i^r + d_i - \text{slack}_{i,j} + M_j^1 \geq 0$. Since $p_i^r, d_i \geq 0$, setting $M_j^1 = \text{slack}_{i,j}$ is valid. On the other hand, in the optimal solution, I_j is 0 if $p_i^r + d_i - \text{slack}_{i,j} \geq 0$, thus ensuring $p_j^r = \max\{0, p_i^r + d_i - \text{slack}_{i,j}\} = p_i^r + d_i - \text{slack}_{i,j}$. Since constraint (16) becomes $p_j^r \leq p_i^r + d_i - \text{slack}_{i,j}$ under $I_j = 0$, to make $p_j^r = p_i^r + d_i - \text{slack}_{i,j}$ in the optimal solution, the right-hand side of constraint (17) should satisfy $M_j^2 \geq p_i^r + d_i - \text{slack}_{i,j}$. Note that by construction, $d_i \leq d_i^*$, $\forall i \in \{1, \ldots, |\mathcal{F}|\}$, $\forall \mathbf{d} \in \mathcal{U}$, thus $p_i^r + d_i - \text{slack}_{i,j} \leq \max\{0, p_i^r + d_i - \text{slack}_{i,j}\} = p_j^r(\mathbf{d}) \leq p_j^r(\mathbf{d}^*)$, $\forall r \in \mathcal{R}_{\mathbf{x}^*}$, $\forall (i,j) \in r$ because $p_j^r(\mathbf{d})$ is a monotonically increasing function. This result shows that $M_j^2 = p_j^r(\mathbf{d}^*)$ is valid. □

Computing tightened big-M according to Proposition 1 requires solving $|\mathcal{F}|$ separate optimization problems. The complexity of these optimization problems depends on how we construct uncertainty set $\mathcal{U}$. Under the set (9) we use, these optimization problems become linear programs, which can be solved very efficiently. With tightened big-M, this formulation works fairly efficiently when we test it for moderately large industry size instances in §4. When using it to solve very large instances, careful construction of uncertainty set $\mathcal{U}$ can greatly speed up (SEP-bigM). For instance, for the uncertainty set described in (8), we could group flights according to their destination airport or pathway and estimate the covariance matrix for each group individually by assuming that primary delays between two flights in two different groups are independent. In that case, $C = \hat{\Sigma}^{-1/2}$ becomes sparse, making (SEP-bigM) easy to solve. Along the same lines, directly estimating a sparse inverse covariance matrix from data is also useful in increasing tractability for large instances (Friedman, Hastie, and Tibshirani 2008; Hsieh et al. 2011).

3.1.2. The Pricing Problem. Each time a new robustifying constraint is added to the relaxed master problem, the resulting program needs to be solved using column generation. Let $G = (\mathcal{N}, \mathcal{A})$ be a directed acyclic graph with a single source node s and a single terminal node t. The node set $\mathcal{N}$ represents the set of flights and arc set $\mathcal{A}$ corresponds to feasible connections between flights. For simplicity and testing purposes, the source node and sink node are dummy nodes that link to all flight nodes. In practice, various operational restrictions could be added. For example, the source node and sink node can only be linked to flights departing from and arriving at maintenance-compatible airports, respectively. Maintenance connection arcs could be added to represent maintenance opportunities (Barnhart et al. 1998a). Each flight node i possesses a weight $-\mu_i$ corresponding to the negative dual price of the ith flight covering constraint (11). We denote by r the dual price of the fleet count constraint (12). Without loss of generality, suppose there are k constraints in the robustifying constraint set (10). Let $\mathbf{d}^1, \mathbf{d}^2, \ldots, \mathbf{d}^k$ be the corresponding flight primary delay vectors and $s_1, s_2, \ldots, s_k$ be the dual prices for robustifying constraints 1 to k. Thus for the RAR pricing problem, we wish to find an $s-t$ path $\pi^* = \{s, n_1, n_2, \ldots, t\}$ that minimizes reduced cost

$$\pi^* = \underset{\pi \text{ is an s-t path}}{\arg\min}\ rc_\pi$$
$$= \underset{\pi \text{ is an s-t path}}{\arg\min} \left\{ \left(-r - \sum_{i \in \pi: i \neq s,t} \mu_i \right) - \sum_{j=1}^{k} \left(\sum_{i \in \pi: i \neq s,t} s_j p_i^\pi(\mathbf{d}^j) \right) \right\}. \tag{18}$$

If reduced cost $rc_{\pi^*} < 0$, the minimizing route π^* forms a new column of A and is assigned a cost of $\sum_{i \in \pi^*: i \neq s,t} p_i^{\pi^*}(\mathbf{d}^j)$ for robustifying constraints $j = 1, 2, \ldots, k$.

3.1.3. Solving the Pricing Problem. We discuss here in detail the revised label-setting algorithm to solve (18), which dynamically calculates the propagated delays using (1) and the reduced cost of the path. As Dunbar, Froyland, and Wu (2012) point out, because the propagated delay $p_i^\pi(\cdot)$ is not a simple sum of delays along the path from s to i, problem (18) cannot easily be cast as a minimum cost network flow problem. The label-setting algorithm we develop here is modified from Dunbar, Froyland, and Wu (2012).

Let π be a $s-t$ path in G (an ordered collection of nodes $\{s, n_1, \ldots, n_q, t\}$ in $\mathcal{N}$ with $(s, n_1), (n_q, t) \in \mathcal{A}$ and $(n_l, n_{l+1}) \in \mathcal{A}$ for all $l = 1, \ldots, q-1$). For $n \in \pi$, let $\pi(n)$ denote the ordered collection of nodes in the path π truncated so that the final node in the list is n. Define truncated reduced cost $rc_{\pi(n)} = (-r - \sum_{i \in \pi(n): i \neq s,t} \mu_i) - \sum_{j=1}^{k} (\sum_{i \in \pi(n): i \neq s,t} s_j p_i^{\pi(n)}(\mathbf{d}^j))$. Denote by $p_n^\pi(\cdot)$ the propagated delay at node n, computed along path $\pi(n)$ using (1). Because of the nonlinear nature of the propagated delay formula (1), our labels must track both the accumulated reduced cost $rc_{\pi(n)}$ at node n along path π and the propagated delay $p_n^\pi(\cdot)$. This motivates the following dominance conditions for paths destined for the same node.

Definition 1 (Path Dominance Condition). Let paths $\pi(n)$, $\eta(n)$ be two different paths destined for the same node n. We say that $\pi(n)$ dominates $\eta(n)$ if $rc_{\pi(n)} \leq rc_{\eta(n)}$, $p_n^{\pi(n)}(\mathbf{d}^j) \leq p_n^{\eta(n)}(\mathbf{d}^j)$, $\forall j \in \{1, \ldots, k\}$ and $(rc_{\pi(n)}, p_n^{\pi(n)}(\mathbf{d}^j)) \neq (rc_{\eta(n)}, p_n^{\eta(n)}(\mathbf{d}^j))$ such that $s_j < 0$, where k is the number of robustifying constraints (10) at the current iteration.

Lemma 1. *Let $m \in \mathcal{N}$ such that $(n, m) \in \mathcal{A}$. If $\pi(n)$ dominates $\eta(n)$, then $\{\pi(n), m\}$ dominates $\{\eta(n), m\}$.*

Proof. From (1) we have $\forall j \in \{1, \ldots, k\}$ such that $s_j < 0$

$$p_m^{\{\pi(n),m\}}(\mathbf{d}^j) = \max\{0, p_n^{\{\pi(n),m\}}(\mathbf{d}^j) + d_n^j - \text{slack}_{n,m}\}, \tag{19}$$

$$p_m^{\{\eta(n),m\}}(\mathbf{d}^j) = \max\{0, p_n^{\{\eta(n),m\}}(\mathbf{d}^j) + d_n^j - \text{slack}_{n,m}\}. \tag{20}$$

By Definition 1, (19), and (20)

$$p_n^{\{\pi(n),\,m\}}(\mathbf{d}^j) = p_n^{\pi(n)}(\mathbf{d}^j) \le p_n^{\eta(n)}(\mathbf{d}^j) = p_n^{\{\eta(n),\,m\}}(\mathbf{d}^j).$$

Thus we have

$$p_m^{\{\pi(n),\,m\}}(\mathbf{d}^j) \le p_m^{\{\eta(n),\,m\}}(\mathbf{d}^j), \quad \forall 1 \le j \le k,\; s_j < 0. \quad (21)$$

For the reduced cost, linear programming duality tells us that dual prices $s_j \le 0$, $\forall 1 \le j \le k$ for all robustifying constraints (10), so we have

$$\begin{aligned} rc_{\{\pi(n),\,m\}} &= rc_{\{\pi(n)\}} - \mu_m - \sum_{j=1}^{k} s_j p_m^{\{\pi(n),\,m\}}(\mathbf{d}^{\mathbf{j}}) \\ &= rc_{\{\pi(n)\}} - \mu_m - \sum_{1\le j\le k:\, s_j<0} s_j p_m^{\{\pi(n),\,m\}}(\mathbf{d}^{\mathbf{j}}), \quad (22) \end{aligned}$$

$$\begin{aligned} rc_{\{\eta(n),\,m\}} &= rc_{\{\eta(n)\}} - \mu_m - \sum_{j=1}^{k} s_j p_m^{\{\eta(n),\,m\}}(\mathbf{d}^{\mathbf{j}}) \\ &= rc_{\{\eta(n)\}} - \mu_m - \sum_{1\le j\le k:\, s_j<0} s_j p_m^{\{\eta(n),\,m\}}(\mathbf{d}^{\mathbf{j}}). \quad (23) \end{aligned}$$

Since $rc_{\{\pi(n)\}} \le rc_{\{\eta(n)\}}$ and (21), we have $rc_{\{\pi(n),\,m\}} \le rc_{\{\eta(n),\,m\}}$. □

By induction, Lemma 1 suggests that if ϖ is a path from m to terminal node t, and $(n, m) \in \mathcal{A}$, we have $rc_{\{\pi(n),\,\varpi\}} \le rc_{\{\eta(n),\,\varpi\}}$. Thus equipped with this definition of path dominance, we can directly apply the label-setting algorithm of Dunbar, Froyland, and Wu (2012) (Algorithm 1) to solve (18) where at each node, we only create labels for those paths that are not dominated by any other paths at that node.

Algorithm 1 (Label-setting algorithm for pricing new columns)
Initialize:
  Set $I_s = \{s\}$ and $I_i = \varnothing$ for all $i \in \mathcal{N}\backslash\{s\}$.
  Set $M_i = \varnothing$ for each $i \in \mathcal{N}$.
loop
  if $\bigcup_{i\in\mathcal{N}}(I_i\backslash M_i) = \varnothing$ **then**
    return $\arg\min_{\pi(t)\in I_t} rc_{\pi(t)}$
  else
    choose $i \in \mathcal{N}$ and $\pi(i) \in I_i\backslash M_i$ so that $rc_{\pi(i)}$ is minimal
    for $(i, j) \in \mathcal{A}$ **do**
      if path $\{\pi(i), j\}$ is not dominated by $\eta(j)$ for any $\eta(j) \in I_j$ **then**
        set $I_j = I_j \cup \{\pi(i), j\}$
      end if
    end for
    set $M_i = M_i \cup \{\pi(i)\}$
  end if
end loop

This label-setting algorithm is fairly efficient on the instances we test in §4. One major reason for this is that some of the robustifying constraints (10) are not binding in the optimal solution of the relaxed master problem. Thus, some of the dual prices corresponding to those robustifying constraints are zero. This makes the pricing algorithm, or more precisely the dominance condition, easier than it appears to be.

3.2. Overall Algorithm

We summarize the column-and-row generation algorithm that we use to solve (epi-RAR) in Algorithm 2. The algorithm iteratively adds robustifying constraints and new columns by solving the corresponding separation and pricing subproblems until the upper bound provided by the separation problem matches with the optimal value of the current relaxed (epi-RAR) problem.

Algorithm 2 (Simultaneous column-and-row generation)
Given:
  Relaxed (epi-RAR) with only constraints (11) and (12)
Initialize:
  Add first robustifying constraint
    $\sum_{r\in\mathcal{R}} \sum_{i\in F(r)} p_i^r(\mathbf{d}^0)x_r \le z$ with an arbitrary $\mathbf{d}^0 \in \mathcal{U}$.
loop
  Solve relaxed (epi-RAR) using branch-and-price, get optimal routes $\mathcal{R}_{\mathbf{x}^*}$ and optimal value z^*.
  Given $\mathcal{R}_{\mathbf{x}^*}$, solve (SEP-bigM), get optimal flight primary delays $\mathbf{d}^*$ and optimal value $z(\mathcal{R}_{\mathbf{x}^*})$.
  if $z(\mathcal{R}_{\mathbf{x}^*}) > z^*$ **then**
    Add robustifying constraint
      $\sum_{r\in\mathcal{R}} \sum_{i\in F(r)} p_i^r(\mathbf{d}^*)x_r \le z$ to relaxed (epi-RAR).
  else
    return $\mathcal{R}_{\mathbf{x}^*}$
  end if
end loop

4. Computational Experiments

In this section, we compare the performance of our robust model against that of the state-of-the-research approach (Dunbar, Froyland, and Wu 2014) on both historical delay data and simulated delay data. All computer programs were written in C++. The branch-and-price scheme was implemented by calling SCIP 3.1.1 (Achterberg 2009) using CPLEX 12.6.0 as the linear programming solver. The mixed-integer program for the separation problem is solved using CPLEX 12.6.0. All computational tests are performed on a laptop equipped with an Intel Core i7-3520M CPU running at 2.9 GHz and 8 GB RAM. For solving some instances in our implementation, we observe that the branch-and-price solution process finds provably good solutions very quickly but fails to prove optimality for a long time. We thus limit the runtime of the branch-and-price process to 120 seconds.

If SCIP failed to prove optimality within 120 seconds, the best dual bound was used to report the optimality gap of the solution. There was no runtime limitation for solving the separation problem.

4.1. Benchmark Model, Local Approach in Dunbar, Froyland, and Wu (2014)

Based on the models and algorithms developed in Dunbar, Froyland, and Wu (2012); Dunbar, Froyland, and Wu (2014) further incorporate stochastic delay information to minimize expected total propagated delay. They model delay stochasticity by constructing a set of random scenarios Ω, where each scenario $\omega \in \Omega$ corresponds to primary delay values $\mathbf{d}^\omega$ for each flight. Each random scenario is realized with equal probability. They develop two algorithms to tackle this problem: (1) the exact approach and (2) the local approach. Since their approaches were originally designed for integrated aircraft routing and crew scheduling, we simplify here to only aircraft routing. The exact approach enumerates all of the feasible aircraft routes $\mathcal{R}$. For each feasible routing $r \in \mathcal{R}$, they calculate its expected total propagated delay $\mathbb{E}_{\mathbf{d}}[c_r(\mathbf{d})] = \sum_{\omega\in\Omega}(1/|\Omega|)(\sum_{i\in F(r)} p_i^r(\mathbf{d}^\omega))$. They then directly solve the deterministic model in (2)–(4) with all decision variables generated. They change the objective function to be $\sum_{r\in\mathcal{R}} \mathbb{E}_{\mathbf{d}}[c_r(\mathbf{d})]x_r$, which represents the sum of expected total propagated delay over all selected routings. Although this is an exact solution approach, it is not practical for industry-sized problems because of its lack of an efficient column generation process. As a result, we consider the local approach as our benchmark for comparison. The local approach is an approximation approach that incorporates stochastic delay information from the scenarios within the label-setting algorithm used in the pricing problem. At each step of the label-setting algorithm, it calculates the average delay propagation arriving at each flight node over all scenarios $\omega \in \Omega$. We denote $\hat{p}_j^r$ as the average propagated delay at node j along path r, and $\hat{p}_j^r = (1/|\Omega|)\sum_{\omega\in\Omega}\max\{0, \hat{p}_i^r + d_i^\omega - \text{slack}_{i,j}\}$. With this notation, we introduce the path dominance condition used in the local approach in Dunbar, Froyland, and Wu (2014).

Definition 2 (Path Dominance Condition in Dunbar, Froyland, and Wu 2014). Let $\pi(n), \eta(n)$ be two different paths destined for the same node n. We say that $\pi(n)$ dominates $\eta(n)$ if $rc_{\pi(n)} \le rc_{\eta(n)}$ and $\hat{p}_n^{\pi(n)} \le \hat{p}_n^{\eta(n)}$.

The optimal solution of the local approach can be computed by column generation using Algorithm 1 with the path dominance condition as in Definition 2.

4.2. Case Studies Based on Historical Delay Data

To evaluate the effectiveness of our robust model, we apply Algorithm 2 to create routes for two of the largest fleet types operated by a major U.S. airline in the year 2007. The characteristics of the underlying networks are listed in Table 1, and all of the flights selected are operated on a daily basis over the testing period. Because in practice the model will be built using historical data and then applied to future operations, all routings were created using ASQP data containing historical flight primary delays for the 31 days of July 2007 (the training set) and then evaluated out-of-sample on the 31 days of August 2007 (the testing set).

Table 1 Characteristics of Two Routing Problems

Network	Number of flight legs	Number of aircraft
N_1	106	24
N_2	117	23

For both sets of data, we benchmark the routes of our robust model against two other routes:

1. The original route in the historical schedule data as the baseline (B): This is the airline's actual route for the selected flights.
2. The local approach provided in Dunbar, Froyland, and Wu (2014) (DFW) as in §4.1: The flight primary delays for the 31 days in July 2007 constitute 31 random delay scenarios. Each random scenario is realized with probability 1/31.

We evaluate the performance of the baseline (B), DFW, and RAR along three criteria: (1) the average total propagated delay, (2) the volatility of total propagated delay (standard deviation), and (3) worst-case total propagated delay (maximum value). Figures 1 and 2 compare the performance of the robust approach over a range of values of the budget of uncertainty $\Gamma \in \{0.2, 0.4, \ldots, 2.8, 3.0\}$ to the DFW method on the training data from July 2007. We compute the relative improvement of DFW and RAR over the baseline (B) for each performance evaluation criterion. More precisely, if we use B, DFW, and RAR to denote the values of a particular criterion of interest for each of the respective approaches, we compute the ratios $100 \times (\text{B-DFW})/\text{B}$ and $100 \times (\text{B-RAR})/\text{B}$. The long dashed lines represent the relative reduction in percentage in corresponding performance criterion for approach DFW over the baseline case (B), and the solid lines represent the relative reduction in percentage in corresponding performance criterion for approach RAR over the baseline (B) under various values of uncertainty budget Γ. Note that for the robust approach, as Γ increases, the performance along the measures of standard deviation and maximum value of propagated delay improves almost monotonically. This is because the routings are both created and tested on training data, and larger values of Γ are more likely to protect against severe propagated delays.

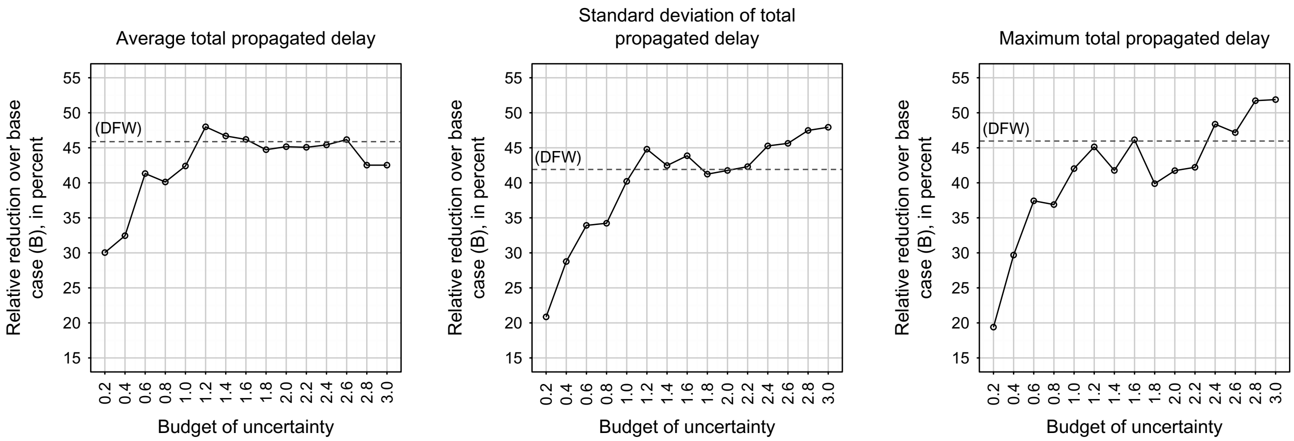

Figure 1 **Relative Improvements of the Algorithms RAR Over B and DFW Over B on Network N_1 (July 2007)**

When the optimal routes informed by the robust model formulated using the training data (July 2007) are applied to testing data (August 2007) as in Figures 3 and 4, we no longer see this monotonicity trend. In practice, we would select one particular value of Γ to create future routings. Guidelines for selecting the budget of uncertainty in previous literature (Bertsimas, Brown, and Caramanis 2011; Bertsimas, Gupta, and Kallus 2013a) usually focus on probabilistic guarantees, i.e., the uncertainty set can be tuned so that constraints are robustly feasible with at least some probability. However, uncertainty of primary delays in our problem does not affect route feasibility, it only changes the amount of total propagated delay. Thus, we decide to choose Γ based on the performance of propagated delay on the training set. We prioritize average total propagated delay as our main performance goal, thus based on Figures 1 and 2, we consider $\Gamma = 1.2$ to be appropriate for network N_1 and $\Gamma = 1.4$ to be appropriate for network N_2 because the robust model yields the lowest average total propagated delay under these two Γ in the training set. Table 2 presents summary statistics for all propagated delay performance criteria under the testing set for routes produced by each approach under both networks N_1 and N_2. Rows with values of Γ that are selected based on training set performance are in bold. Note that generally, the best way to set the parameter Γ would be to (1) divide our historical data set into training and validation components; (2) select the value of Γ that yields the best performance on the validation set (cross-validation can also be applied); and (3) once that value of Γ is set, we would then use it to generate future aircraft routings. For the sake of simplicity and also because of a dearth of data, we only use the training set to select the value of Γ in this work.

Both approaches improve on the baseline routing considerably in terms of average, standard deviation, and maximum of total propagated delay. For testing data on network N_1, when $\Gamma = 1.2$, RAR performs better than DFW in all of the performance criteria of total propagated delay that we consider (8.2% larger decrease in average total propagated

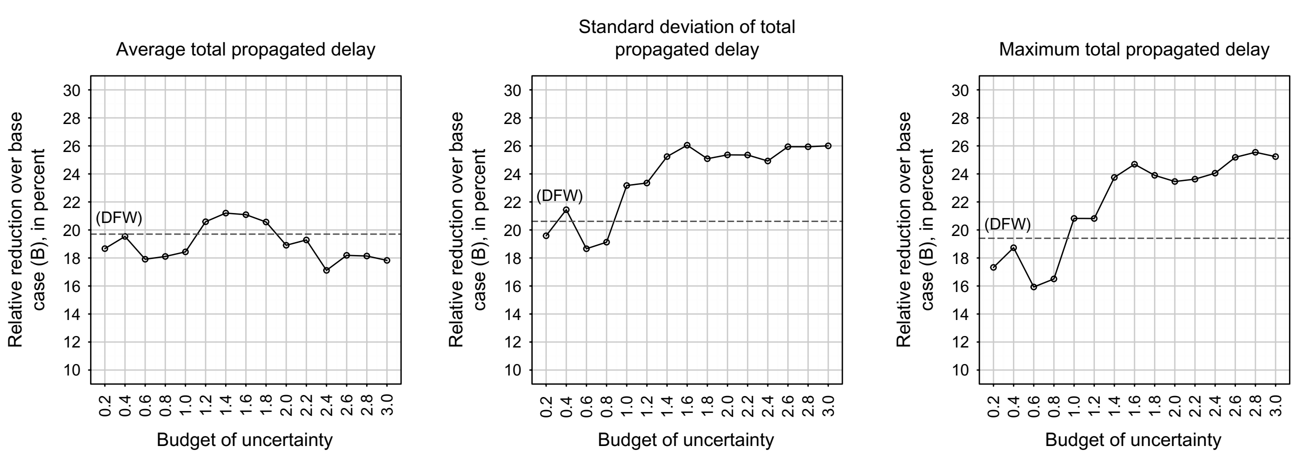

Figure 2 **Relative Improvements of the Algorithms RAR Over B and DFW Over B on Network N_2 (July 2007)**

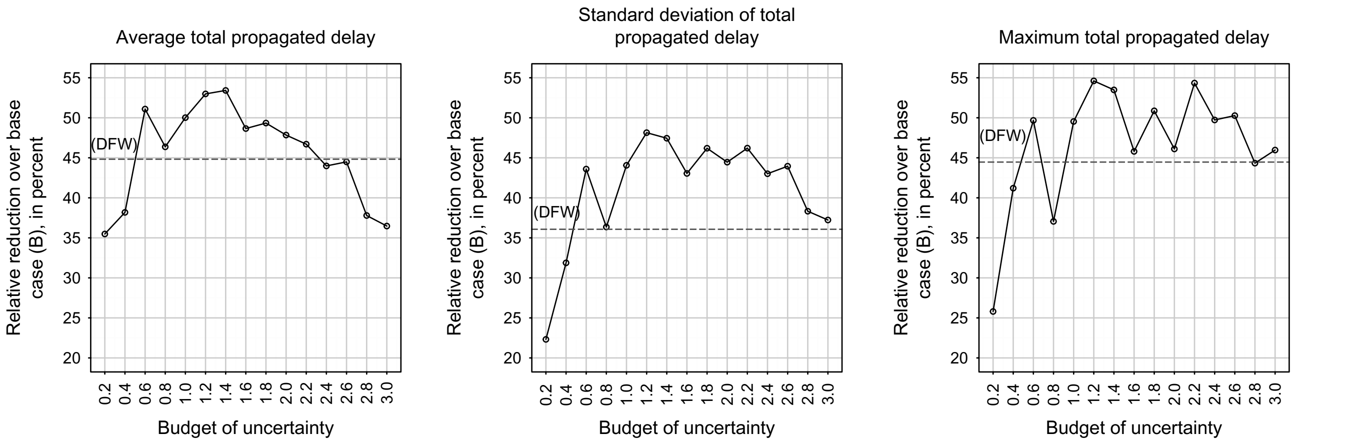

Figure 3 **Relative Improvements of the Algorithms RAR Over B and DFW Over B on Network N_1 (August 2007)**

delay, 12.1% larger decrease in standard deviation, and 10.1% larger decrease in maximum total propagated delay, as compared to the baseline). For testing data on network N_2, when $\Gamma = 1.4$, RAR outperforms DFW according to the standard deviation (2.2% larger decrease from baseline) and the maximum value of total propagated delay (3.6% larger decrease from baseline), but loses slightly in terms of average total propagated delay (1.0% smaller decrease from baseline).

In addition to this specific value that we have inspected, for both flight networks under testing data, RAR actually outperforms DFW in reducing volatility and extreme delay under a wide range of Γ values (for network N_1, $\Gamma \in [1.0, 2.6]$; for network N_2, $\Gamma \in [0.2, 1.8]$). For very small (≤ 0.4) or very large (≥ 2.6) Γ values, RAR loses superiority in at least two evaluation criteria in one of the networks. This is because the uncertainty set $\mathcal{U}$ constructed by these Γ are either too conservative (for large Γ values) or too restrictive (for small Γ values) in representing potential primary delay. In other words, such Γ values do not accurately capture real-world data. When $\Gamma \in [1.0, 2.6]$, in network N_1, compared to DFW, RAR can reduce on average 9.1% more of the standard deviation and on average 6.1% more of the maximum delay value from the baseline. When $\Gamma \in [0.2, 1.8]$, in network N_2, RAR can reduce on average 2.1% more of the standard deviation and on average 5.0% more of the maximum delay value compared to DFW from the baseline. The superiority in reducing extreme delay values of RAR intuitively makes sense because of the min-max objective in the RAR formulation. The reduction in volatility is a surprising by-product of reduced extreme delay values, even though we do not explicitly include it in the objective for RAR. Most surprising, for network N_1, RAR also has a better performance in reducing average total propagated delay in the training set. When $\Gamma \in [1.0, 2.6]$, on average 3.8% more of the average total propagated delay can be reduced. In network N_2, RAR mostly underperforms DFW in reducing average delay value, but

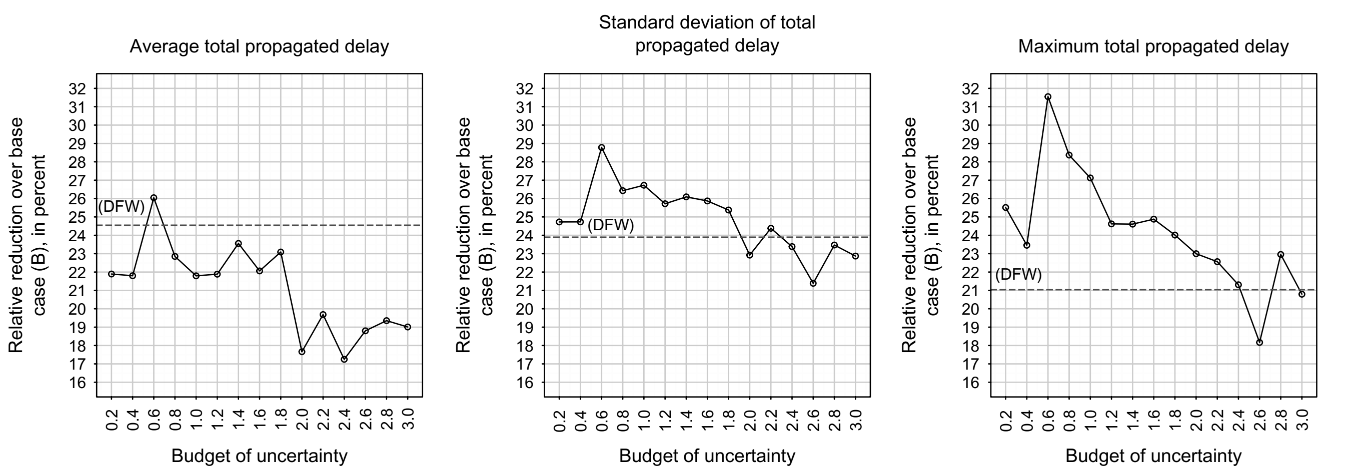

Figure 4 **Relative Improvements of the Algorithms RAR Over B and DFW Over B on Network N_2 (August 2007)**

Table 2 Comparative Algorithmic Performance on Two Test Networks

Approach	Avg. total propagated delay (minute)	Std. total propagated delay (minute)	Max total propagated delay (minute)	Total CPU time (seconds)	SEP-bigM CPU time (seconds)	Number of robustifying cuts added	Optimality gap (%)
			Flight network N_1, 106 flights/24 aircraft				
B	1,042.6	1,409.8	5,856	—	—	—	
DFW	575.3	901.2	3,252	217.04	—	—	0.00
RAR ($\Gamma = 0.2$)	672.6	1,095.2	4,345	161.70	3.56	3	3.79
RAR ($\Gamma = 0.4$)	644.5	1,023.7	3,874	679.01	49.65	6	1.06
RAR ($\Gamma = 0.6$)	509.5	795.3	2,947	572.90	53.75	5	2.33
RAR ($\Gamma = 0.8$)	559.2	897.3	3,686	223.63	61.30	3	2.41
RAR ($\Gamma = 1.0$)	521.1	788.6	2,955	403.16	109.15	5	0.39
RAR ($\Gamma = 1.2$)	**490.2**	**731.1**	**2,658**	**602.34**	**185.94**	**7**	**0.00**
RAR ($\Gamma = 1.4$)	485.7	740.9	2,724	1,041.99	491.78	5	0.93
RAR ($\Gamma = 1.6$)	535.3	802.9	3,174	935.79	505.36	5	0.24
RAR ($\Gamma = 1.8$)	528.2	758.5	2,877	2,004.52	1,064.69	8	0.62
RAR ($\Gamma = 2.0$)	543.8	783.1	3,156	779.50	337.32	4	0.43
RAR ($\Gamma = 2.2$)	555.7	758.4	2,674	1,482.82	898.21	5	0.45
RAR ($\Gamma = 2.4$)	584.0	803.4	2,944	4,123.59	3,146.46	8	1.49
RAR ($\Gamma = 2.6$)	578.7	790.2	2,912	3,315.71	2,453.99	7	2.00
RAR ($\Gamma = 2.8$)	648.6	869.5	3,260	1,758.19	918.81	6	1.75
RAR ($\Gamma = 3.0$)	662.3	884.9	3,164	1,593.94	1,102.16	6	1.32
			Flight network N_2, 117 flights/23 aircraft				
B	1,131.9	1,651.4	7,392	—	—	—	
DFW	854.0	1,256.7	5,837	140.11	—	—	0.00
RAR ($\Gamma = 0.2$)	884.1	1,243.1	5,506	28.43	2.38	3	0.00
RAR ($\Gamma = 0.4$)	885.2	1,243.0	5,658	23.68	3.97	2	0.00
RAR ($\Gamma = 0.6$)	837.1	1,176.1	5,060	420.06	18.50	5	0.01
RAR ($\Gamma = 0.8$)	873.3	1,214.9	5,295	50.73	5.86	2	0.00
RAR ($\Gamma = 1.0$)	885.2	1,210.1	5,387	320.87	22.26	4	0.09
RAR ($\Gamma = 1.2$)	884.2	1,226.7	5,572	77.06	12.93	3	0.00
RAR ($\Gamma = 1.4$)	**865.3**	**1,220.5**	**5,573**	**209.96**	**17.89**	**4**	**0.00**
RAR ($\Gamma = 1.6$)	882.2	1,224.2	5,553	335.99	22.93	4	0.00
RAR ($\Gamma = 1.8$)	870.5	1,232.3	5,617	321.51	11.92	4	0.09
RAR ($\Gamma = 2.0$)	932.0	1,272.9	5,692	489.12	12.49	4	0.05
RAR ($\Gamma = 2.2$)	909.1	1,248.8	5,724	420.46	10.27	4	0.09
RAR ($\Gamma = 2.4$)	936.7	1,265.2	5,817	446.10	10.50	4	0.06
RAR ($\Gamma = 2.6$)	919.1	1,298.2	6,049	95.82	9.73	2	0.00
RAR ($\Gamma = 2.8$)	912.8	1,263.8	5,695	94.95	8.71	2	0.00
RAR ($\Gamma = 3.0$)	916.7	1,273.7	5,855	100.52	5.68	2	0.00

not by too much (on average only −1.8% when $\Gamma \in [0.2, 1.8]$), especially in light of the gains made in reducing volatile and extreme values.

As for computational tractability, the efficiency of the RAR model depends on the tractability of both the separation problem and the branch-and-price problem. Table 2 reports the total central processing unit (CPU) time, CPU time for the separation problem, number of robustifying cuts needed until Algorithm 2 converges, and the optimality gap for the final robust solution. Since we impose 120 seconds runtime limit on the branch-and-price solution process, the best objective value we get from each branch-and-price process cannot serve as a valid lower bound for the optimal value of RAR if SCIP fails to prove optimality. However, the best dual bound at the end of the solution process can still provide a valid lower bound of RAR. We denote UB as the lowest optimal objective value of all of the separation problems when Algorithm 2 converges. UB serves as an upper bound for the optimal value of RAR. Similarly, we denote LB as the highest dual bound value of all of the branch-and-price processes solved when Algorithm 2 terminates. LB is a valid lower bound for the optimal value of RAR. The optimality gap is then defined as $(\text{UB} - \text{LB})/\text{UB} \times 100\%$. Within the runtime limitation, for network N_1, RAR only solves one instance to optimality ($\Gamma = 1.2$), however, for many instances, provably good solutions (with optimality gap less than 1%) can be obtained; for network N_2, RAR solves 9 out of 15 instances to optimality, and all of the unsolved instances have very high-quality solutions (with optimality gap less than 0.1%). We observe that when the separation problem and branch-and-price process are both easy to solve (e.g., the nine fully solved instances in network N_2), the RAR model is more efficiently solved compared to DFW. However, when one of the problems is relatively hard to solve, RAR is less efficient to solve. Computational time reduction for the separation problem could be achieved by performing matrix sparsifying tricks suggested in §3.1.1. Tailored branching rules (Barnhart et al. 1998a) could be utilized to speed up the branch-and-price solution process. We also point out here that the solution time of RAR does not depend on the size of the historical delay data for training. No matter how many days of historical flight delay we use, we can transform all of them into a single uncertainty set of the same size. On the other hand, the solution time of DFW grows approximately linearly with the number of days involved in the training set because the algorithm needs to average over all training days for each flight leg during the column generation process. This suggests that RAR might require less computational effort when our model uses more historical delay data as part of its training set.

In this case study, careful modeling of the uncertainty set $\mathcal{U}$ allows us to reduce the volatility of solution performance. In some cases, average performance also benefits when compared to the existing state-of-the-research stochastic optimization approach. We suspect that these benefits come from the following: (1) the robust approach provides additional robustness when the realized primary delays are slightly different from the estimated distribution in the training data set; and (2) the robust model provides a tractable and exact approach to deal with correlated primary delays, and incorporating delay correlation improves the robustness of resultant routings.

4.3. Case Studies Based on Simulated Delay Data

To provide more insight into the relative performance of RAR over DFW, we conduct an additional round of computational experiments where the flight primary delay data are generated from simulated probability distributions instead of historical data. We seek to quantify the robustness of both methods with respect to changes in probability distributions. These case studies are motivated by the idea that primary flight leg delays rarely corresponds to a specific distribution, but rather a composite of several types of delays, each with differing individual distributions that may vary throughout different times of the day, month, and year (Tu, Ball, and Jank 2008). Because of this distribution ambiguity and variation over time, it is an important feature of robust operations planning methods to be able to protect against ambiguity of future delays when trained on historical data.

Previous literature suggests several classes of candidate probability distributions that are commonly used to model flight leg delay (Mueller and Chatterji 2002; Tu, Ball, and Jank 2008; Bai 2006). Out of these, we pick three representative ones: normal (truncated), gamma, and log normal. The computational experiment is set up as follows. For the training data set, we use historical flight delay data in July 2007 (the training set we use in §4.2) to compute the observed mean and variance of each flight leg and the Spearman's rank correlation coefficient matrix of all flight legs. We pick Spearman's rho instead of the widely used Pearson's rho because Spearman's rho preserves order when nonlinear transformations are applied to random variables. This is convenient for generating random variables with different distributions, but the same correlation structure (MathWorks 2014). We then generate the set of training data with 1,000 samples for each flight leg such that the marginal distributions of flight legs follow a truncated normal distribution with the same mean and variance calculated from the July 2007 data. We also fix the Spearman's rank correlation coefficient matrix of the simulated data to be the same as the one calculated from July 2007 data. For testing data sets, we create three different groups of testing sets where the marginal distributions of flight leg delay are distributed as truncated normal, gamma, and log normal, respectively. For each group, we create testing sets in the following way:

- *Deviation in mean*

We keep the standard deviation and Spearman's rank correlation coefficient matrix of the testing set the same as the training set. We then generate testing data by setting the mean to be 0.5, 0.75, 1, 1.25, 1.5, 1.75, and 2 times the mean of the training set.

- *Deviation in standard deviation*

We keep the mean and Spearman's rank correlation coefficient matrix of the testing set the same as the training set. We then generate testing data by setting the standard deviation of the testing set to be 0.5, 0.75, 1, 1.25, 1.5, 1.75, and 2 times the standard deviation of the training set.

- *Deviation in correlation structure*

We keep the mean and standard deviation of the testing set the same as the training set. We then generate testing data with different Spearman's rank correlation coefficient matrix. We define the multiplier of correlation $\alpha \in (-\infty, 1]$ as a parameter to control how the correlation structure deviates from the training set. To be specific, $\alpha = 0$ means there is no change in correlation structure. As α increases to 1, the primary delays become more independent, with full independence at $\alpha = 1$. As α decreases to $-\infty$, the primary delay data becomes more correlated, with perfect correlation at $\alpha = -\infty$. We pick $\alpha = \ln(3/4)$, $\ln(5/6)$, $\ln(11/12)$, 0, $\ln(1+(e-1)/12)$, $\ln(1+(e-1)/6)$, $\ln(1+(e-1)/4)$ to generate testing data with various correlation structures. The detailed meaning of α and the correlation perturbing procedures are provided in the online appendix (available as supplemental material at http://dx.doi.org/10.1287/trsc.2015.0657).

In total, we have 19 testing sets for each distribution and each flight network. We apply the above generation procedure to both networks N_1 and N_2 that we used in §4.2. We use the training set to create routings using both RAR and DFW and then test the performance of the generated routings under the testing sets created using the procedure above. Note that in DFW, there are 1,000 random scenarios, each realized with probability 1/1,000. Similar to §4.2, for both networks, we select uncertainty budget Γ that has the best performance in reducing average total propagated delay in the training set. For network N_1, $\Gamma = 1.2$; for network N_2, $\Gamma = 1.4$. Since we do not have a baseline routing anymore, we calculate the relative performance ratio of RAR over DFW as $100 \times (\text{DFW} - \text{RAR})/\text{DFW}$ for each performance criteria. The results are summarized in Tables 3–5, where each table corresponds to a specific testing data delay distribution. Complete results with all values of $\Gamma \in \{0.2, 0.4, \ldots, 2.8, 3.0\}$ are presented in the online appendix.

Overall, in all 114 testing sets, RAR outperforms DFW in reducing average value in 58% (66/114) of the cases, reducing standard deviation in 99% (113/114) of the cases, and reducing extreme value in 99% (113/114) of the cases.

When testing data take the same distributional form as the training data (truncated normal, Table 3), for both networks, RAR consistently outperforms DFW in reducing standard deviation and extreme value, especially when testing data have a larger mean or standard deviation than the training set. The only exception is the case when the standard deviation multiplier is 0.5 in network N_2. On the other

Table 3 **Relative Performance Ratio (RPR) Under Different Mean/Standard Deviation/Correlation Multiplier of Testing Data Following Truncated Normal Distribution**

Multiplier of mean	Multiplier of std	Multiplier of correlation	RPR of mean reduction (%)	RPR of std reduction (%)	RPR of extreme value reduction (%)
		Flight network N_1, 106 flights and 24 aircraft, best $\Gamma = 1.2$			
0.50	1.00	0.00	−0.5	4.1	0.4
0.75	1.00	0.00	−0.3	4.3	0.1
1.00	1.00	0.00	−0.1	4.5	1.3
1.25	1.00	0.00	0.2	4.7	1.9
1.50	1.00	0.00	0.5	4.9	2.8
1.75	1.00	0.00	0.9	5.1	3.2
2.00	1.00	0.00	1.2	5.2	3.4
1.00	0.50	0.00	−5.2	3.4	2.8
1.00	0.75	0.00	−1.6	4.1	0.3
1.00	1.25	0.00	1.0	4.9	3.8
1.00	1.50	0.00	1.9	5.3	2.5
1.00	1.75	0.00	2.5	5.7	2.6
1.00	2.00	0.00	3.0	5.9	2.5
1.00	1.00	$\ln(3/4)$	0.7	5.7	6.7
1.00	1.00	$\ln(5/6)$	0.4	5.2	6.7
1.00	1.00	$\ln(11/12)$	0.0	4.0	1.7
1.00	1.00	$\ln(1+(e-1)/12)$	−0.3	4.0	5.4
1.00	1.00	$\ln(1+(e-1)/6)$	−0.3	4.3	2.9
1.00	1.00	$\ln(1+(e-1)/4)$	−0.2	4.7	10.2
		Flight network N_2, 117 flights and 23 aircraft, best $\Gamma = 1.4$			
0.50	1.00	0.00	−2.4	1.2	1.1
0.75	1.00	0.00	−2.0	1.5	1.2
1.00	1.00	0.00	−1.6	1.8	1.0
1.25	1.00	0.00	−1.2	2.1	0.9
1.50	1.00	0.00	−0.9	2.4	1.0
1.75	1.00	0.00	−0.6	2.6	1.3
2.00	1.00	0.00	−0.3	2.8	1.3
1.00	0.50	0.00	−5.5	−1.1	1.7
1.00	0.75	0.00	−3.0	1.1	1.1
1.00	1.25	0.00	−1.2	2.2	1.4
1.00	1.50	0.00	−0.1	2.4	1.6
1.00	1.75	0.00	0.3	2.5	1.1
1.00	2.00	0.00	0.6	2.6	1.0
1.00	1.00	$\ln(3/4)$	−2.1	1.4	1.3
1.00	1.00	$\ln(5/6)$	−2.3	1.0	3.5
1.00	1.00	$\ln(11/12)$	−2.1	1.2	1.2
1.00	1.00	$\ln(1+(e-1)/12)$	−1.9	0.9	2.3
1.00	1.00	$\ln(1+(e-1)/6)$	−1.7	1.7	0.7
1.00	1.00	$\ln(1+(e-1)/4)$	−2.3	0.8	2.4

Table 4 Relative Performance Ratio (RPR) Under Different Mean/Standard Deviation/Correlation Multiplier of Testing Data Following Gamma Distribution

Multiplier of mean	Multiplier of std	Multiplier of correlation	RPR of mean reduction (%)	RPR of std reduction (%)	RPR of extreme value reduction (%)
		Flight network N_1, 106 flights and 24 aircraft, best $\Gamma = 1.2$			
0.50	1.00	0.00	3.9	9.2	20.4
0.75	1.00	0.00	3.2	8.4	18.4
1.00	1.00	0.00	2.8	7.9	17.0
1.25	1.00	0.00	2.5	7.6	16.0
1.50	1.00	0.00	2.3	7.3	15.1
1.75	1.00	0.00	2.1	7.2	13.8
2.00	1.00	0.00	2.2	7.0	10.7
1.00	0.50	0.00	−2.4	5.9	13.5
1.00	0.75	0.00	1.3	7.4	16.0
1.00	1.25	0.00	3.6	8.2	17.9
1.00	1.50	0.00	3.9	8.3	18.4
1.00	1.75	0.00	4.0	8.3	18.8
1.00	2.00	0.00	4.1	8.4	19.2
1.00	1.00	$\ln(3/4)$	3.5	7.3	9.1
1.00	1.00	$\ln(5/6)$	4.0	7.9	9.4
1.00	1.00	$\ln(11/12)$	1.4	4.0	8.2
1.00	1.00	$\ln(1+(e-1)/12)$	1.6	6.6	8.0
1.00	1.00	$\ln(1+(e-1)/6)$	3.1	7.0	1.2
1.00	1.00	$\ln(1+(e-1)/4)$	3.2	8.5	10.8
		Flight network N_2, 117 flights and 23 aircraft, best $\Gamma = 1.4$			
0.50	1.00	0.00	2.2	9.4	27.9
0.75	1.00	0.00	1.2	4.7	15.6
1.00	1.00	0.00	0.6	3.9	11.0
1.25	1.00	0.00	0.2	3.5	9.9
1.50	1.00	0.00	0.0	3.3	9.0
1.75	1.00	0.00	0.0	3.3	8.4
2.00	1.00	0.00	0.0	3.4	7.9
1.00	0.50	0.00	−3.6	0.8	5.8
1.00	0.75	0.00	−0.9	2.9	9.4
1.00	1.25	0.00	1.5	4.6	14.9
1.00	1.50	0.00	2.1	5.2	15.7
1.00	1.75	0.00	2.7	5.6	15.8
1.00	2.00	0.00	3.1	6.0	15.8
1.00	1.00	$\ln(3/4)$	−0.5	2.4	1.0
1.00	1.00	$\ln(5/6)$	−1.0	2.2	6.6
1.00	1.00	$\ln(11/12)$	−0.3	2.8	2.9
1.00	1.00	$\ln(1+(e-1)/12)$	−0.4	1.7	1.2
1.00	1.00	$\ln(1+(e-1)/6)$	−0.3	3.0	1.4
1.00	1.00	$\ln(1+(e-1)/4)$	−0.5	1.8	2.3

hand, in most cases, DFW outperforms RAR in reducing average propagated delay, especially when the testing data have a smaller mean or variance compared to the training data and when the testing data have a different correlation structure. Overall, similar to what we observe in the computational results of network N_2 in §4.2, the improvement in reducing volatility and extreme value comes at the cost of a diminished performance with regard to decreasing the average value. However, this cost is usually small in comparison to the reductions in volatility and extreme value that we observe.

When the testing set is distributed as gamma or log normal (i.e., different from the training set, Tables 4 and 5), in most of the cases when only mean and standard deviation are varying, RAR results in a route that has a substantially smaller standard deviation, extreme value, and even average value. As before, the increase in average total propagated delay is relatively small. When the correlation structure deviates from the testing data, we see that the impact of such deviation on RAR's performance is much larger than the impact of deviation in mean or standard deviation. Under almost all correlation multipliers, all three performance criteria (especially for the extreme value reduction criterion) are inferior to the cases when correlation structure is unchanged. Deviation in correlation changes the shape of the uncertainty set instead of the volume, which might cause this substantial downgrade in RAR's performance. Also, changing correlation structure might have less impact on the performance of DFW because the approach

Table 5 Relative Performance Ratio (RPR) Under Different Mean/Standard Deviation/Correlation Multiplier of Testing Data Following Log-Normal Distribution

Multiplier of mean	Multiplier of std	Multiplier of correlation	RPR of mean reduction (%)	RPR of std reduction (%)	RPR of extreme value reduction (%)
Flight network N_1, 106 flights and 24 aircraft, best $\Gamma = 1.2$					
0.50	1.00	0.00	3.8	11.1	25.8
0.75	1.00	0.00	2.8	10.0	24.3
1.00	1.00	0.00	2.3	9.5	23.1
1.25	1.00	0.00	2.0	9.1	22.1
1.50	1.00	0.00	1.8	8.8	21.2
1.75	1.00	0.00	1.7	8.5	20.4
2.00	1.00	0.00	1.7	8.2	19.5
1.00	0.50	0.00	−2.0	9.5	21.2
1.00	0.75	0.00	1.0	9.4	22.5
1.00	1.25	0.00	3.0	9.5	23.7
1.00	1.50	0.00	3.4	9.5	24.1
1.00	1.75	0.00	3.7	9.5	24.5
1.00	2.00	0.00	3.9	9.6	24.9
1.00	1.00	$\ln(3/4)$	1.8	5.3	8.2
1.00	1.00	$\ln(5/6)$	2.6	5.6	5.8
1.00	1.00	$\ln(11/12)$	−1.0	1.5	6.9
1.00	1.00	$\ln(1+(e-1)/12)$	0.2	5.1	5.6
1.00	1.00	$\ln(1+(e-1)/6)$	1.4	3.7	−4.6
1.00	1.00	$\ln(1+(e-1)/4)$	2.2	7.2	8.9
Flight network N_2, 117 flights and 23 aircraft, best $\Gamma = 1.4$					
0.50	1.00	0.00	1.5	8.5	15.3
0.75	1.00	0.00	0.6	7.3	15.4
1.00	1.00	0.00	0.0	6.3	15.3
1.25	1.00	0.00	−0.4	5.6	15.0
1.50	1.00	0.00	−0.4	5.2	14.7
1.75	1.00	0.00	−0.4	5.0	14.5
2.00	1.00	0.00	−0.3	4.8	14.2
1.00	0.50	0.00	−3.6	3.3	12.8
1.00	0.75	0.00	−1.3	5.3	14.7
1.00	1.25	0.00	0.8	6.9	15.5
1.00	1.50	0.00	1.4	7.4	15.6
1.00	1.75	0.00	1.8	7.7	15.6
1.00	2.00	0.00	2.2	8.0	15.6
1.00	1.00	$\ln(3/4)$	−1.6	2.3	0.8
1.00	1.00	$\ln(5/6)$	−2.3	2.0	6.5
1.00	1.00	$\ln(11/12)$	−1.1	2.7	2.5
1.00	1.00	$\ln(1+(e-1)/12)$	−1.5	1.6	1.2
1.00	1.00	$\ln(1+(e-1)/6)$	−1.4	2.6	0.0
1.00	1.00	$\ln(1+(e-1)/4)$	−1.5	1.8	4.7

does not explicitly use correlation data. Though RAR is less effective in these correlation-varying cases, it still outperforms DFW in at least two criteria. Moreover, the small inferiority in one criterion, if it exists, is usually offset by the superiority of the other two criteria.

In summary, compared to the existing stochastic optimization approach, the RAR model provides additional benefit when realized flight leg delay differs in distribution from historical data.

5. Concluding Remarks and Future Research Directions

In this paper, we propose a robust optimization-based formulation for the aircraft routing problem to minimize the maximal possible total propagated delay, assuming flight primary leg delays live in a prespecified uncertainty set. We provide data-driven methods to construct a flight leg delay uncertainty set that not only includes the uncertainty of individual flights but also the correlations among different flights. We then reformulate the robust problem as an integer program. We propose an exact decomposition solution approach under a column-and-row generation framework. Importantly, this solution method can be applied to general robust optimization problems where the nominal problem is solved through branch and price.

Using real-world instances with actual schedule and delay data as well as simulated delay data, we demonstrate the effectiveness and efficiency of our method.

We show that by incorporating delay correlation, our robust model outperforms the state-of-the-research stochastic optimization approach in reducing all three performance criteria considered: average value, standard deviation, and maximum value of total propagated delay in most of the cases. In the cases when deficit in one criterion exists, such inferiority is usually offset by gains in the other two criteria.

We attribute these benefits to characteristics of our solution method: (1) the robust approach provides a tractable and exact method for dealing with correlated uncertainty, thus providing additional robustness against propagated delay; and (2) the robust approach is less vulnerable to the variations in the input data when compared to stochastic optimization approaches (i.e., the robust approach provides additional robustness when the realized flight delays distributionally differ from historical data).

The idea of applying robust optimization to robust airline planning is fairly new. There are many possible future research directions in this area. For example, a direct extension of this work could be to apply robust optimization to the integrated aircraft routing and crew pairing problem. It may also be interesting to investigate tailored approaches to modeling flight delay uncertainty sets, so as to achieve greater tractability as well as probabilistic performance guarantees. We believe that robust optimization, with its merits of tractability and attractive solution quality, represents a promising direction for dealing with uncertainty in airline scheduling problems.

Supplemental Material

Supplemental material to this paper is available at http://dx.doi.org/10.1287/trsc.2015.0657.

Acknowledgments

The authors want to thank Professors Cynthia Barnhart and Dimitris Bertsimas at the Massachusetts Institute of Technology and Professor Vikrant Vaze at Dartmouth College for their valuable comments and suggestions. The authors also thank the referees for several suggestions that improved the manuscript. This material is based on work (for the second author) supported by the National Science Foundation Graduate Research Fellowship [Grant 1122374]. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Achterberg T (2009) SCIP: Solving constraint integer programs. *Math. Programming Comput.* 1(1):1–41.

Ahmadbeygi S, Cohn A, Lapp M (2010) Decreasing airline delay propagation by re-allocating scheduled slack. *IIE Trans.* 42(7):478–489.

Bai Y (2006) Analysis of aircraft arrival delay and airport on-time performance. Unpublished master's thesis, University of Central Florida, Orlando.

Barnhart C, Bertsimas D, Caramanis C, Fearing D (2012) Equitable and efficient coordination in traffic flow management. *Transportation Sci.* 46(2):262–280.

Barnhart C, Johnson EL, Nemhauser GL, Savelsbergh MWP, Vance PH (1998b) Branch-and-price: Column generation for solving huge integer programs. *Oper. Res.* 46(3):316–329.

Barnhart C, Boland NL, Clarke LW, Johnson EL, Nemhauser GL, Shenoi RG (1998a) Flight string models for aircraft fleeting and routing. *Transportation Sci.* 32(3):208–220.

Ben-Tal A, Nemirovski A (1999) Robust solutions of uncertain linear programs. *Oper. Res. Lett.* 25(1):1–13.

Bertsimas D, Sim M (2004) The price of robustness. *Oper. Res.* 52(1):35–53.

Bertsimas D, Thiele A (2006) A robust optimization approach to inventory theory. *Oper. Res.* 54(1):150–168.

Bertsimas D, Brown DB, Caramanis C (2011) Theory and applications of robust optimization. *SIAM Rev.* 53(3):464–501.

Bertsimas D, Dunning I, Lubin M (2016) Reformulations versus cutting planes for robust optimization. *Comput. Management Sci.* 13(2):195–217.

Bertsimas D, Gupta V, Kallus N (2013a) Data-driven robust optimization. arXiv preprint arXiv:1401.0212.

Bertsimas D, Pachamanova D, Sim M (2004) Robust linear optimization under general norms. *Oper. Res. Lett.* 32(6):510–516.

Bertsimas D, Litvinov E, Sun XA, Zhao J, Zheng T (2013b) Adaptive robust optimization for the security constrained unit commitment problem. *Power Systems, IEEE Trans.* 28(1):52–63.

Borndörfer R, Dovica I, Nowak I, Schickinger T (2010) Robust tail assignment. *Proc. Fiftieth Ann. Sympos. AGIFORS, Nice, France.*

Bureau of Transportation Statistics (2013) Research and innovative technology. U.S. Department of Transportation, Washington DC, http://www.transtats.bts.gov/ot_delay/OT_DelayCause1.asp?pn=1.

Dunbar M, Froyland G, Wu C-L (2012) Robust airline schedule planning: Minimizing propagated delay in an integrated routing and crewing framework. *Transportation Sci.* 46(2):204–216.

Dunbar M, Froyland G, Wu C-L (2014) An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing. *Comput. Oper. Res.* 45:68–86.

Federal Aviation Administration (2014) Aviation maintenance technician handbook—general. U.S. Department of Transportation, Washington, DC, https://www.faa.gov/regulations_policies/handbooks_manuals/aircraft/amt_handbook/.

Friedman J, Hastie T, Tibshirani R (2008) Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9(3):432–441.

Froyland G, Maher SJ, Wu C-L (2014) The recoverable robust tail assignment problem. *Transportation Sci.* 48(3):351–372.

Hsieh C-J, Dhillon IS, Ravikumar PK, Sustik MA (2011) Sparse inverse covariance matrix estimation using quadratic approximation. *Adv. Neural Inform. Processing Systems* 24:2330–2338.

Kang LS (2004) Degradable airline scheduling: An approach to improve operational robustness and differentiate service quality. Unpublished doctoral thesis, Massachusetts Institute of Technology, Cambridge, MA.

Klabjan D, Schaefer AJ, Johnson EL, Kleywegt AJ, Nemhauser GL (2001) Robust airline crew scheduling. *Proc. TRISTAN IV, Azores, Portugal.*

Lan S, Clarke J-P, Barnhart C (2006) Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Sci.* 40(1):15–28.

Marla L, Barnhart C (2010) Robust optimization: Lessons learned from aircraft routing. Working paper, Massachusetts Institute of Technology, Cambridge, MA.

MathWorks (2014) Simulating dependent random variables using copulas. http://www.mathworks.com/help/stats/examples/simulating-dependent-random-variables-using-copulas.html#zmw57dd0e179.

Mueller ER, Chatterji GB (2002) Analysis of aircraft arrival and departure delay characteristics. *AIAA's Aircraft Technology, Integration and Operations (ATIO) Technical Forum, Los Angeles*.

Rosenberger JM, Johnson EL, Nemhauser GL (2004) A robust fleet-assignment model with hub isolation and short cycles. *Transportation Sci.* 38(3):357–368.

Shebalov S, Klabjan D (2006) Robust airline crew pairing: Move-up crews. *Transportation Sci.* 40(3):300–312.

Smith BC, Johnson EL (2006) Robust airline fleet assignment: Imposing station purity using station decomposition. *Transportation Sci.* 40(4):497–516.

Tu Y, Ball MO, Jank WS (2008) Estimating flight departure delay distributions—a statistical approach with long-term trend and short-term pattern. *J. Amer. Statist. Assoc.* 103(481):112–125.

Weide O, Ryan D, Ehrgott M (2010) An iterative approach to robust and integrated aircraft routing and crew scheduling. *Comput. Oper. Res.* 37(5):833–844.

Yen JW, Birge JR (2006) A stochastic programming approach to the airline crew scheduling problem. *Transportation Sci.* 40(1):3–14.