Robust Optimization: Lessons Learned from Aircraft Routing

Lavanya Marla
Massachusetts Institute of Technology, lavanya@mit.edu

Cynthia Barnhart
Massachusetts Institute of Technology, barnhart@mit.edu

Building robust airline scheduling models involves constructing schedules and routes with reduced levels of both flight delays and passenger and crew disruptions. This might be achieved by building plans that require fewer recovery actions, or that decrease the complexity or cost of recovery. In this paper, we study different classes of models to achieve robust airline scheduling solutions. We focus on the aircraft routing problem, a step of the airline scheduling process. In particular, we study two general classes of robust models and one tailored approach that uses domain knowledge to guide the solution process. The general classes of models are (i) extreme-value based, and (ii) chance-constrained or probability-based. We first show how the general models can be applied to the aircraft routing problem. Then, based on model structure and tractability considerations, we suggest extensions to the general models, capturing domain knowledge and reducing the need to iterate and re-solve the models. The extended models and insights gleaned in this work apply not only to the aircraft routing model but also to the broad class of large-scale, network-based, resource allocation. We provide extended formulations for our extended models in the appendices. We discuss how general and tailored models are similar or different with respect to modeling, complexity and tractability, and we compare solution performance with respect to the robustness metrics of interest.

Key words: airline scheduling, aircraft routing, robust routing, robust optimization

1. Introduction

Historically, airline schedule planners modeled the air transportation system assuming that flights can depart and arrive as scheduled. Factors such as weather and air traffic delays, however, regularly affect the Department of Transportation's (DoT) (US DoT 2009) on-time performance of the system, as evidenced by the 15-minute flight on-time arrival rate in 2008 of 70.55%, an all-time low in the 2000-2008 period (Bureau of Transportation Statistics 2009a).

Air travel has increased tremendously in the past decade. 1450 billion passenger revenue kilometers were traveled in North America in 2007. An even higher growth has been observed in the past five years in the Asia-Pacific regions (International Civil Aviation Organization December 2007), (International Air Transport Association 2007). With these high growth rates, the percent of US aircraft arriving late has increased from 17.01% in 2003 to 25.96% in 2008 (Bureau of Transportation Statistics 2009a). These delays are highly detrimental in an industry where the profit margins are typically less than 2%, with delay and fuel costs forming a major component of operating costs (Air Traffic Association 2008). As a case in point, 29% of U.S. airline operating cost incurred in 2007 was due to fuel (Air Traffic Association 2008) and the total costs of U.S. domestic air traffic delays crossed the $40 billion mark (CNNMoney.com May 22, 2008). Of these, $19.1 billion represents incremental operating costs due to delays for the airlines (including additional fuel costs of $1.6 billion, releasing 7.1 million metric tons of carbon dioxide into the atmosphere), $12 billion represents the estimated passenger costs due to low productivity and lost business, and $10 billion represents losses to other industries that rely on air traffic (CNNMoney.com May 22, 2008). It is evident that if delays can be reduced, society and the airline industry would benefit tremendously.

Robust airline scheduling is a way of *pro-actively* considering such delays and disruptions and creating schedules with the objective of building plans that are less susceptible to disturbances or easier to repair once disrupted. This is in contrast to prior practice, where responses to delays were *reactive*, that is, after an event occurred, schedule recovery actions which can be costly and complex to implement, were taken. Robust airline scheduling, then, is a proactive planning technique aimed at reducing total *realized* costs, including both plan and recovery costs.

To evaluate the robustness of solutions obtained, we use simulation, as the objective function values of the planning optimization models do not indicate the *realized costs* or *robustness* of the solution. Instead, we evaluate the solution through a *simulation* and measure its performance with respect to a host of relevant robustness metrics.

1.1. Motivation

In this paper, our focus is on the aircraft routing step of the airline scheduling process. The aircraft routing problem is to find a feasible sequence of flight legs, called aircraft routings or rotations, to be operated by each aircraft so that maintenance restrictions on aircraft are satisfied. Each flight is required to be assigned to (or covered by) exactly one aircraft, using no more than the available number of available aircraft. and meeting all maintenance requirements. Though robust planning is required at every step of the airline scheduling process, we choose aircraft routing because of its high impact on schedule reliability and relatively low impact on crew costs, flight costs and passenger revenues (Lan et al. 2006).

We demonstrate how aircraft routings differ and what we mean by robust aircraft routings, with an example. In Table 1, we report performance for 7 aircraft routings as measured by the percent of flights in the routing that arrive within 15 minutes, 30 minutes, 60 minutes, 120 minutes and 180 minutes of their respective scheduled arrival times. Note that these percentages were calculated over 22 days of operations of a major U.S. airline. For the instances under consideration in this paper, all of which are drawn from actual airline operations, we compare metrics of interest, as detailed in §3.1. The reported variability in flight delays is significant, as even small differences in the range of 1% can improve/deteriorate the airline's ranking with respect to the DoT's 15-minute on-time performance metric (Bureau of Transportation Statistics 2009c). Because airlines do not typically explicitly consider delays in selecting aircraft routings, the airline effectively might choose at random any of these routings, and thus, can incur high delays. To illustrate, for this instance, the aircraft routing operated by the airline is Routing 5, with DoT on-time performance ranking third from the bottom. Moreover, in addition to aircraft delay disparities, different routings can lead to different levels of *passenger* disruptions and delays. A passenger is considered to be disrupted if one or more flight legs on his itinerary are canceled, or if delays cause insufficient connection time to the next flight leg in his/her itinerary. The percentage of passengers disrupted '% D-pax reduced' relative to the airline's routing are shown in Table 1. Routings 1 and 2 can vastly improve upon the airline's routing without any additional resources, while Routing 7 can deteriorate the airline's performance greatly.

	Flight Delays					Pax Disruptions	
Routing	≤15 min	≤30 min	≤60 min	≤120 min	≤180 min	#D-pax	% D-pax reduced
Routing 1	79.1	86.7	93.4	98	99.1	988	10.14
Routing 2	78.8	86.8	93.2	98.2	99.2	986	10.3
Routing 3	78.3	86.2	92.9	98.1	99	1028	6.5
Routing 4	78.3	86	92.9	97.5	98.6	1047	4.8
Routing 5	77.7	85.8	92.8	97.7	98.9	1100	0
Routing 6	77.6	85.7	92.4	97.4	98.6	1057	3.9
Routing 7	76.5	84.7	92	97.2	98.5	1223	-11.2

Table 1 Flight Delay Percentages and Passenger Disruptions of Feasible Routings, N_2

The relationship between aircraft routings and delays and disruptions can be explained by the phenomenon of *propagated delays*. In network structures, flight delays can be divided into two components (Lan et al. 2006): *independent delays* that originate at the flight's origin or during the flight, and *propagated delays* resulting from delays in upstream flights that are not absorbed by slack time between flight legs. Delay propagation is illustrated in Fig. 1. The solid arrows show the planned schedule for flights f_1 and f_2; and the dotted arrows the operated schedule. PDT, ADT, PAT and AAT are the planned departure time, actual departure time, planned arrival time and actual arrival time respectively, of flight f_2. In Figure 1, flight f_1 is delayed, and its delay causes the remaining slack time between flight f_1's arrival and f_2's scheduled departure to be less than the minimum connection time required for the same aircraft to fly both flight legs. This causes propagated delay PD for flight leg f_2. In addition, independent delay is incurred by f_2, both at its departure (IDD) and its arrival (IAD), resulting in total departure delay (TDD) and total arrival delay (TAD). However, by changing the sequence of flights operated by each aircraft, propagated delay can be reduced (Figure 2). This involves changes in the routings of aircraft, but because *no* new aircraft are being added and the flight schedule remains unchanged, the total slack in the system is not altered, instead only the positioning of the slack is changed.

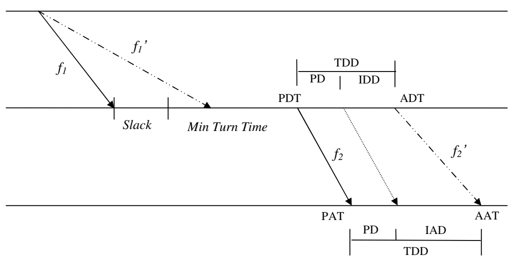


Figure 1 **Delay Propagation along an Aircraft Route**

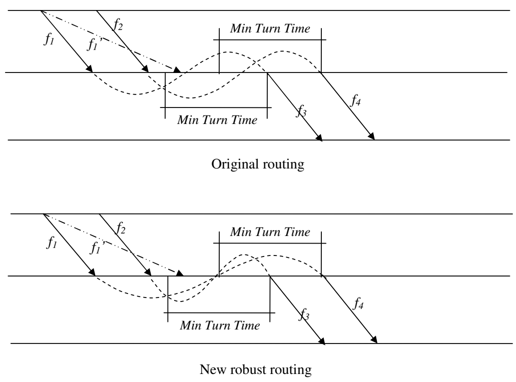


Figure 2 **Robust Routing with Optimal Slack Allocation**

For the airline we study, propagated delay typically represents 20% to 30% of total flight delay (Lan et al. 2006). Because total independent delay is a constant for the flight schedule, reducing delay propagation by choosing Routing 1 instead of, for example, Routing 7 has the effect of reducing *total* delay. The differences in propagated delays for Routings 1-7 are shown in Fig 3. Different aircraft routings are not very different on 'good' days like Day 10, but differences become apparent on 'bad' days like Day 5.

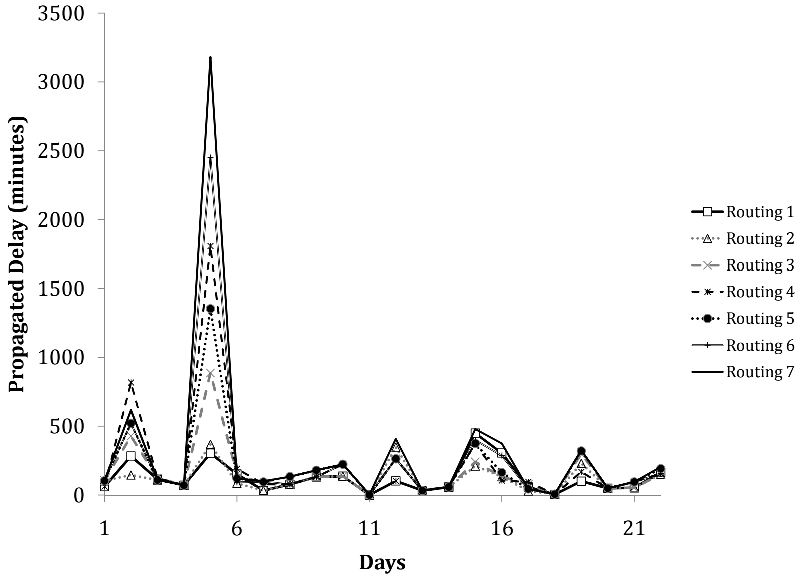


Figure 3 Propagated Delays of Feasible Aircraft Routings, N_2

1.1.1. Discussion of Metrics Ideally, good solutions to airline scheduling problems ensure low levels of delay for flights, and good travel experience with low passenger delays and disruptions. Metrics, then, such as total flight delay minutes, total cost of delay, 15-minute on-time performance, 30-min on-time performance, and 60-minute on-time performance, are all examples of measures that reflect airline schedule reliability and robustness. A difficulty, however, is that these metrics are not always aligned with each other. For example, the 15-minute on-time performance metric does not reflect delays greater than 15 minutes, and therefore maximizing 15-minute on-time performance is not the same as minimizing total delay minutes. Similarly, minimizing aircraft delay minutes is different from minimizing passenger delay minutes or passenger disruptions because fewer passengers can be disrupted by holding flights to allow passengers to make their connections, thus increasing total aircraft and passenger delays.

1.2. Contributions

In this paper, we study three different approaches to robustness in aircraft routing, and hence, airline scheduling - two that are generally applicable, the extreme-value based approach and a probabilistic chance-constrained programming approach; and one that is a tailored approach proposed by Lan, Clarke and Barnhart (Lan et al. 2006). The extreme-value approach considered is the robust optimization approach of Bertsimas and Sim (Bertsimas and Sim 2004), (Bertsimas and Sim 2003) and the probabilistic approach is the Chance-Constrained Programming approach of Charnes and Cooper (Charnes and Cooper 1959), (Charnes and Cooper 1963). We begin by showing how to model the robust aircraft routing problem using these three approaches, and identify their respective limitations; suggesting extensions and enhancements to the models to address these limitations. We then evaluate the similarities and differences in models and solutions generated by these different approaches, using a simulation-based evaluator. The findings and extensions from this work are generally applicable to the broad class of network-based resource allocation problems.

1.3. Outline

In §2, we present the three classes of robust models we study. We present Charnes and Cooper's Chance-Constrained Programming approach, the robust optimization approach of Bertsimas and Sim, and Lan, Clarke and Barnhart's robust aircraft routing approach. In addition, we propose extensions and enhancements to the general classes of models in §2. We present the experimental set-up for our computations, and details of the simulator built to evaluate the performances of the models in §3. In §4, we compare the models and solutions generated by the different approaches in terms of complexity and run times (§4.1), model parameters (§4.2) and modeling paradigms (§4.3), and show how robust solutions may be generated by all classes of models. In §5, we summarize our findings.

2. Robust Models

2.1. The Standard Deterministic Aircraft Routing Model

Following is the standard deterministic aircraft maintenance routing formulation, denoted AR, which we attempt to make robust in following sections.

$$
\begin{aligned}
&\mathbf{AR}: \\
&\min 0 && && (2.1)\\
&\text{s.t.} \sum_{s\in S} a_{is}x_s = 1 && \forall\, i \in F && (2.2)\\
&\sum_{s\in S_i^+} x_s - y_{i,d}^- + y_{i,d}^+ = 0 && \forall\, i \in F^+ && (2.3)\\
&-\sum_{s\in S_i^-} x_s - y_{i,a}^- + y_{i,a}^+ = 0 && \forall\, i \in F^- && (2.4)\\
&\sum_{s\in S} r_s x_s + \sum_{g\in G} p_g y_g \le N && && (2.5)\\
&y_g \ge 0 && \forall\, g \in G && (2.6)\\
&x_s \in \{0,1\} && \forall\, s \in S && (2.7)
\end{aligned}
$$

The decision variables in this formulation correspond to strings (sequences of flight legs) with each string starting and ending at a maintenance station and obeying FAA and other regulatory rules regarding maximum time between maintenance. Strings capture multiple decisions simultaneously, and thus are *composite variables* (Armacost et al. 2002). These are typically used when they yield strong formulations, and/or when they remove the need to include complex, difficult to model, constraints, as in the case of aircraft routing.

Let F be the set of daily flights, F^+ be the set of flight legs originating at a maintenance station, F^- be the set of flight legs ending at a maintenance station, and S be the resulting set of possible strings. The set of ground arcs (including wraparound arcs, beginning in one day and ending in another day at the same location) is denoted by G, the set of flight legs ending with flight leg i is S_i^-, and the set of flight legs beginning with flight leg i is S_i^+. Ground variable $y_{i,d}^-$ represents the number of aircraft on the ground before flight leg i departs and $y_{i,d}^+$ is the number of aircraft on the ground after flight leg i departs, for all flight legs i. Similarly $y_{i,a}^-$ is the number of aircraft on the ground before flight leg i arrives and $y_{i,a}^+$ is the number of aircraft on the ground after flight leg i arrives, for all flight legs i. a_{is} is 1 if flight leg $i \in F$ is contained in string $s \in S$ and 0 otherwise. r_s is the number of times string $s \in S$ crosses the count line, p_g is the number of times ground arc $g \in G$ crosses the count line, and N is the number of aircraft available.

Constraints (2.2) are the cover constraints that require each flight leg to be covered exactly once. Constraints (2.3) and (2.4) balance the number of aircraft at each location and constraints (2.5) count the number of aircraft. x_s takes on value 1 if string $s \in S$ is selected to be operated by an aircraft, and 0 otherwise. y_g is the number of aircraft on ground arc $g \in G$.

2.2. Modeling

The first challenges associated with achieving robust solutions are deciding what constitutes robustness and how to capture these in the optimization model. In the following sections, we illustrate how different modeling paradigms will force different modeling approaches resulting in different solutions and different computational challenges.

Using schedule data and details of operated aircraft routings (typically available from the ASQP database (Bureau of Transportation Statistics 2009b)), Lan, Clarke and Barnhart compute the independent and propagated delay for each flight leg. Because independent delays are, by definition, independent of aircraft routings, independent delays can be applied to any sequence of flights forming a string to estimate propagated delay (PD_i), total departure delay ($TDD(i)$) and total arrival delay (TAD_i) for each flight leg i along that string, or along *any* possible string, even those not operated by the airline. With this, it is also possible to compute the probability of a flight leg being delayed to a certain extent, or the probability of a string experiencing a specified level of propagated delay, or the range of delays experienced by a flight leg or a string. These characterizations of uncertainty are employed in the following sections.

2.3. Tailored Approach

Lan, Clarke and Barnhart (Lan et al. 2006) attempt to make an aircraft routing robust by driving AR using the metric of total expected propagated delay. Recall from §1.1, that total propagated delay differs from total aircraft delay by a constant value. Thus, minimizing expected propagated delay is equivalent to minimizing expected total aircraft delay.

Let pd^s_{ij} be the propagated delay from flight leg i to flight leg j when i immediately precedes j in string s. Using the notation introduced in 2.1, and assuming that the different strings in the network are independent of each other, the expected propagated delay in the network is:

$$E\left[\sum_{s\in S}\left(\sum_{(i,j)\in s} pd^s_{ij}\right)x_s\right] = \sum_{s\in S}\left(x_s \times \sum_{(i,j)\in s} E[pd^s_{ij}]\right) = \sum_{s\in S} d_s x_s \tag{2.8}$$

where $d_s = \sum_{(i,j)\in s} E[pd^s_{ij}]$.

In terms of problem structure and complexity, the Lan, Clarke and Barnhart tailored model (LCB) is the same as AR, except that the feasibility objective is replaced with the objective of minimizing total expected propagated delay, specifically:

$$\mathbf{LCB}:$$
$$\min \sum_{s\in S} d_s x_s \tag{2.9}$$
$$\text{s.t. Cover, Balance, Count, and Integrality } (2.2)-(2.7) \tag{2.10}$$

2.4. Probabilistic Chance-Constrained Programming Approach

In Chance-Constrained Programming, the chance or probability that a constraint of the model is satisfied is required to exceed some specified threshold level. The chance-constrained formulation of aircraft routing is as follows:

$$\max 0 \tag{2.11}$$
$$\text{s.t. } \mathbf{P}\Big(\sum_{s\in S} a_{is}x_s = 1\Big) \geq \alpha_i \qquad \forall\, i \in F \tag{2.12}$$
$$\text{Balance, Count, and Integrality} (2.3)-(2.7), \tag{2.13}$$

where $\mathbf{P}$ denotes probability, and α_i is a user-defined 'protection' parameter indicating the minimum probability that the aircraft routing solution will satisfy constraint i. Charnes and Cooper

(Charnes and Cooper 1959) describe how to model the non-linear constraints of (2.12) as linear constraints.

We model uncertainty in the flight cover constraints (2.2) by defining p_{is} as the probability obtained from historical data, that flight leg i in string s is covered, that is, is delayed by fewer than t minutes when string s is operated. We let $t = 90$ minutes to indicate the threshold beyond which flight cancelations, and hence flight non-coverage, are likely to occur. Because each flight leg is present only in a single string in the solution, the probability of flight i being delayed less than t minutes is $p_i = \sum_{s \in S} p_{is} x_s$. Using this, we re-write (2.11) - 2.13 as:

$$
\begin{aligned}
&\mathbf{CCP}: \\
&\quad \max 0 && && (2.14)\\
&\quad \text{s.t.} \sum_{s \in S} a_{is} x_s = 1 && \forall\, i \in F && (2.15)\\
&\qquad \sum_{s \in S} p_{is} x_s \geq \alpha_i && \forall\, i \in F && (2.16)\\
&\qquad \text{Balance, Count, and Integrality} (2.3) - (2.7)
\end{aligned}
$$

Constraints (2.16) are the 'robustness constraints' that limit to α_i the probability that flight leg i is delayed more than t minutes in the operation of string s.

In structure and complexity, CCP is similar to AR, though it adds constraints (2.16), one for each flight leg, and requires specification of the value of α_i for each constraint $i \in F$. A challenge associated with this model is the specification of α values. Too high values can lead to infeasibilities and too low can lead to inadequate levels of robustness or protection. As a result, CCP might have to be solved repeatedly to find appropriate α-values. Repeated solution of CCP, however, might be both impractical and ineffective in identifying the best α-values. To overcome these limitations of CCP, we develop the $\alpha - CCP$ model (2.17) - (2.20), in which the protection levels α for constraints (2.16) need not be specified apriori and instead are decision variables in the model. The objective of $\alpha - CCP$ (2.17) is to maximize the sum of protection levels of all the constraints.

$$
\begin{aligned}
&\alpha - \mathbf{CCP}: \\
&\quad \max \sum_{i \in F} \alpha_i && && (2.17)\\
&\quad \text{s.t.} \sum_{s \in S} a_{is} x_s = 1 && \forall\, i \in F && (2.18)\\
&\qquad \alpha_i \leq \sum_{s \in S} p_{is} x_s && \forall\, i \in F && (2.19)\\
&\qquad \text{Balance, Count, and Integrality (2.3) – (2.7)} && && (2.20)
\end{aligned}
$$

Alternative objective functions include: 1) maximizing a weighted sum of flight probabilities $\left(\sum_{i\in F} p_i w_i\right)$ with weight w_i assigned to flight i; or 2) maximizing the minimum probability α_i $(\max \alpha_{min})$ with additional constraints $\alpha_{min} \leq \sum_{s\in S} p_{is} x_s$, for all $i \in F$. We present our general formulation for the $\alpha - CCP$, applicable to all linear programs, called the Extended Chance-Constrained Formulation, in Appendix C.

2.5. Extreme-Value Robust Optimization Approach

We adapt the extreme-value robust optimization approach of Bertsimas and Sim to the aircraft routing problem by letting $\hat{a}_{is} = -1$ if flight $i \in F$ in string $s \in S$ has extreme value of delay exceeding t minutes, based on historical data. Because the Bertsimas and Sim approach considers realizations of the uncertain parameters at their extreme (or worst-case) values ((Bertsimas and Sim 2004), (Bertsimas and Sim 2003)), if a flight i in string s has extreme delay exceeding t minutes using historical data, the extreme value $\hat{a}_{is} = -1$ results in $a_{is} + \hat{a}_{is} = 1 - 1 = 0$, reflecting the extreme occurrence that flight i is canceled, and hence, uncovered.

In the cover constraint for flight $i \in F$, let S_i be the set of strings $s \in S$ whose coefficients a_{is} are subject to uncertainty. For each flight i, the Bertsimas and Sim robust optimization approach defines a 'robustness' or 'protection' parameter Γ_i taking on (possibly continuous) values in $[0, |S_i|]$, representing the number of coefficients in the solution in constraint i that can assume worst-case or extreme values and still satisfy feasibility of the constraint. The model ensures constraint feasibility in the case when up to $\lfloor \Gamma_i \rfloor$ coefficients take on extreme values, and one coefficient a_{it} changes by $(\Gamma_i - \lfloor \Gamma_i \rfloor)\hat{a}_{it}$. For our purposes, integer values of Γ_i are the most meaningful because of the uncertainty definition. Γ_i represents the number of strings, for each flight $i \in F$, in which flight i cannot experience delays greater than or equal to t minutes in the extreme case. The resulting extreme value formulation from the Bertsimas and Sim model is:

$$\min 0 \tag{2.21}$$

$$\text{s.t.} \sum_{s\in S} a_{is} x_s + \max_{\{S_i' \cup t_i | S_i' \subseteq S_i, |S_i'| = \lfloor \Gamma_i \rfloor, t_i \in S_i \setminus S_i'\}} \left\{ \sum_{s \in S_i} \hat{a}_{is} w_s + (\Gamma_i - \lfloor \Gamma_i \rfloor) \hat{a}_{it_i} w_t \right\} = 1 \qquad \forall\, i \in F \tag{2.22}$$

$$x_s \leq w_s \qquad \forall\, s \in S \tag{2.23}$$

$$w_s \geq 0 \qquad \forall\, s \in S \tag{2.24}$$

$$\text{Balance, Count, and Integrality}(2.3) - (2.7) \tag{2.25}$$

If $\Gamma_i = 0$ then the cover constraint for flight i reduces to $\sum_{s \in S} a_{is} x_s = 1$, with robustness concerns effectively ignored. If $\Gamma_i \geq 1$, the constraint ensures that each flight $i \in F$ is covered by at least one string s that both contains flight i and has extreme value of delay less than t for i. EV ensures that each flight is covered, in the worst-case, by placing it in more than one string if needed, thus *over-covering* flights. Because the values of $\hat{a}_{is}$ are either 0 or -1, (2.21)-(2.25) simplifies to:

$$\mathbf{EV}:$$

$$\min 0 \tag{2.26}$$

$$\text{s.t.} \sum_{s \in S} a_{is} x_s + \max \left\{ \sum_{s \in S} \hat{a}_{is} x_s, -\Gamma_i \right\} = 1 \tag{2.27}$$

$$\text{Balance, Count, and Integrality} (2.3) - (2.7), \tag{2.28}$$

with the second term in (2.27) representing the protection level, and can be linearized easily.

Note that the level of robustness can be varied by selecting different values of the extreme delay parameter t. Reducing the value of t will have the effect of generating more conservative solutions, those that permit little delay, while increasing the value of t will have the opposite effect. This change in t produces a similar effect in the CCP and $\alpha - CCP$ as well.

As in the CCP model, the EV model requires the specification of a parameter value (in this case, Γ_i) for each cover constraint (2.27). To avoid the need to repeatedly solve EV to determine the best Γ-values, likely an impractical exercise, we propose an alternate model, denoted $\Delta - EV$, in which Γ parameters are modeled as decision variables. This model is a tailored formulation of our general *Delta* model that we present in Appendix B. The goal of this formulation is to minimize, for each constraint i, the number of variables in the solution whose coefficients are *not* allowed to realize their extreme values (denoted by Δ_i), in order to ensure feasibility of the constraint. (Note that Δ_i is the converse of Γ_i.) We minimize sum of Δ_i for all flight legs i, thus maximizing the protection level. Due to the special structure, the $\Delta - EV$ model can be formulated without any Δ variables, as follows:

$$\mathbf{\Delta - EV}:$$

$$\max \sum_{i \in F} \sum_{s \in S} \hat{a}_{is} x_s \tag{2.29}$$

$$\text{s.t.} \sum_{s \in S} a_{is} x_s \geq 1 \qquad \forall i \in F \tag{2.30}$$

$$\text{Balance, Count, and Integrality} (2.3) - (2.7) \tag{2.31}$$

The objective (2.29) effectively maximizes $\sum_{i\in F}\Gamma_i$ or minimizes $\sum_{i\in F}\Delta_i$ by maximizing the total value of the protection function in (2.27) summed for all flights $i \in F$.

2.5.1. Capturing Uncertainty in the Objective Because the extreme value robust optimization framework also allows uncertainty to be modeled in the objective function, an alternative to EV and $\Delta - EV$ is to capture uncertainty in a manner similar to that of the Lan, Clarke and Barnhart tailored LCB model, but within the extreme value framework. This can be translated into limiting the total amount of propagated delay when any Γ strings in the solution experience their respective extreme propagated delay values and all other strings experience their nominal value of propagated delay (that is, zero). This results in a formulation denoted the $Obj - EV$ model, with a complex protection function like (2.22). The detailed formulation is presented in the Appendix A. The difficulty with this model, again, is specifying the 'best' value of Γ. Because a value of Γ does not indicate the overall robustness of the solution, it is difficult to specify apriori. We therefore present the $\Delta - Obj - EV$ model for which we find a solution such that the maximum propagated delay D exceeds the sum of the extreme delay values of any subset of Γ strings, with Γ maximal. The $\Delta - Obj - EV$ solution, therefore, is an aircraft routing that allows the largest number of strings to realize their worst-case propagated delay values without exceeding the maximum allowable system propagated delay D. This model, like the $\Delta - EV$, is a simplified version of our general Delta model, presented in Appendix B.

D is a threshold on total propagated delay, obtained by analyzing the historical occurrences of propagated delays. A possible value of D could be 50% of the average total propagated delay on a bad day. We set to zero the nominal propagated delay value for any string s, and let $\hat{d}_s$ represent the expected extreme or worst-case value of propagated delay for string s, as computed using historical data. Let $\bar{S}$ be the set of strings $s \in S$ with realizations of non-zero propagated delays in the historical data, that is, with non-zero $\hat{d}_s$. We set $v_s = 1$ if string $s \in S$ has to take on its nominal propagated delay value of 0 and not its worst-case (historical) value for the solution to be feasible. To maximize the size of the minimal subset of strings that can realize their worst-case values and ensure feasibility, we sort the strings in increasing order of their $\hat{d}_s$ values such that $\hat{d}_1 \leq \hat{d}_2 \leq ... \leq \hat{d}_{|S|}$. Δ is the maximum number of strings with propagated delays subject to uncertainty that must assume nominal, not worst-case, values to be feasible.

$\mathbf{\Delta - Obj - EV}$:

$$\min \Delta \tag{2.32}$$

$$\text{s.t.} \sum_{s \in S} \hat{d}_s x_s - \sum_{s \in S} \hat{d}_s v_s \leq D \tag{2.33}$$

$$\Delta \geq \sum_{s \in S} v_s \qquad \forall\, s \in \bar{S} \tag{2.34}$$

$$v_s \leq x_s \qquad \forall\, s \in \bar{S} \tag{2.35}$$

$$v_s \leq w_s \qquad \forall\, s \in \bar{S} \tag{2.36}$$

$$v_s \geq x_s + w_s - 1 \qquad \forall\, s \in \bar{S} \tag{2.37}$$

$$w_s \geq w_{s+1} \qquad \forall\, s \in |S| - |\bar{S}| + 1, ..., |S| - 1 \tag{2.38}$$

$$w_{|S|-|\bar{S}|+1} \leq 1 \qquad \forall\, i \in I \tag{2.39}$$

$$w_{|S|} \geq 0 \qquad \forall\, i \in I \tag{2.40}$$

Cover, Balance, Count, Integrality

$$(2.2) - (2.7) \tag{2.41}$$

$$v_s \in [0, 1] \qquad \forall\, s \in S \tag{2.42}$$

$$w_s \in \{0, 1\} \qquad \forall\, s \in S \tag{2.43}$$

Constraints (2.33) require that the total worst-case propagated delay be limited by D when any $|\bar{S}| - \Delta$ strings take on worst-case propagated delay values. $v_s = 1$ for all strings $s \in \bar{S}$ *in the solution* whose delay value *must* be set to 0 to achieve feasibility. $w_s = 1$ for all strings $s \in \bar{S}$ for which there exists a $k \geq s$ such that $v_k = 1$. Inequalities (2.35) force $v_s = 0$ unless string s is present in the solution. Inequalities (2.36) allow v_s to be 1 only if w_s is 1. (2.37) allow v_s to be 1 only if both $w_s = 1$ and $x_s = 1$. Constraints (2.34), in combination with the objective (2.32) count the maximum number of strings $s \in \bar{S}$ whose coefficients must take on the nominal propagated delay value of 0. The explanation for this constraint lies in the realization that when the strings in S are sorted in increasing order of $\hat{d}_s$ values, the *maximum number* of coefficients that must assume nominal values to maintain feasibility is determined by forcing the strings with the smallest $\hat{d}_s$ values, for $s \in \bar{S}$, to have their associated v_s values set to 1 if x_s is in the solution. Constraints (2.38), (2.39) and (2.40) require w_s for $s \in \bar{S}$ to form a step function, so that the maximal set of w_s can be set to 1. Constraints (2.42) and (2.43) restrict w and v variables to take on values of 0 or 1 only.

3. Evaluation

3.1. Experimental Set-up

We conduct our experiments using data representing the flight network of a major US airline that operates a hub-and-spoke network. We identify the sub-networks for two different aircraft types, with each representing a different aircraft routing problem, denoted N_1 and N_2. The schedules of both networks are daily schedules, that is, the set of flight legs is operated every day by each aircraft type. The characteristics of N_1 and N_2 are shown in Table 2. Historical flight leg delay and cancelation data are obtained from the Airline Service Quality Performance (ASQP) database (Bureau of Transportation Statistics 2009b) for two of the busiest months of the year. Both the months have similar schedules, load factors, and levels of delay. Delay data consisting of 19 days in the first month (referred to as historical data) is used to derive delay information (distributions and expected values) of flights. These data are used as inputs to the aircraft routing models. The solutions are then evaluated using delay data for 22 days of the following month (referred to as future data). Our models are implemented in C++ and OPL Studio v6.0 on a Dell PC with 1 GB RAM.

In evaluating robust routings, we assume that flight delays are allowed to propagate along the string, without any recovery interventions such as cancelations or swaps. This allows us to estimate the levels of delay propagation and robustness of the strings that may occur *without* intervention. Because cancelation and swapping strategies are different for different airlines, assuming particular swapping and cancelation strategies could lead to bias.

Fleet Type	Daily Flights	Locations
N_1	38	10
N_2	50	16

Table 2 Fleet Network Characteristics

3.2. Metrics and Simulator

We assess the robustness of an aircraft routing solution by the following metrics:

1. Expected on-time performance for all legs in the flight schedule for 15 minutes, 60 minutes, 120 minutes, and 180 minutes;
2. Total expected number of passenger disruptions; and
3. Total expected daily flight delay.

These metrics reflect DoT performance and passenger-centric metrics. Because total flight delay is equal to propagated delay plus independent delay (which is constant), comparison results related to propagated delay also apply to total delay.

For our simulator, we use Lan, Clarke and Barnhart's algorithm to compute from the airline's solution, for each day in the 'future' month of data, the propagated delay ($PD(i)$) and independent delay ($ID(i)$) from the total departure and arrival delays $TDD(i)$ and $TAD(i)$ for each flight i. Following that, we also compute metrics such as total flight delay for each day, 15 minute on-time performance, 30 minute on-time performance, and the total number of passengers disrupted for the airline's routing. To compute passenger disruptions, we enumerate all pairs of flights f_1, f_2

between which passengers connect. Let $C(f_1, f_2)$ be the scheduled time available for the passenger to make the connection and let $m_{f_1 f_2}$ represent the minimum time needed for a passenger to connect between flights f_1 and f_2. Then, if $C(f_1, f_2) - TAD(f_1) + TDD(f_2) \leq m_{f_1 f_2}$, the actual connection time between f_1 and f_2 is too short and passengers scheduled to make this connection are disrupted. Then, given independent delay for each day in the 'future' data for each flight leg i, we re-employ Lan, Clarke and Barnhart's algorithm (in the reverse order of steps) to compute the same delay and disruption metrics for each of the solutions we generate using our models. Note that the models use historical data to generate the solution, and then the 'future data' is used to evaluate them.

4. Results

In this section, we present the results of our experiments, studying similarities and differences in the solutions obtained in terms of complexity, run time and robustness as measured by our metrics.

4.1. Typical Computation Times

Table 3 reports average computation times for the airline instances solved in this work.

Model	Parameters	Iterations	Run time per iteration
AR	None	1	5 sec
LCB	None	1	10 sec
CCP	$\alpha_i \forall i$	53	7 sec
$\alpha - CCP$	None	1	7 sec
EV	$\Gamma_i \forall i$	50	35 sec
$Obj - EV$	Γ	15	45 sec-10 hrs (sometimes out of memory)
$\Delta - EV$	None	1	28 sec
$\Delta - Obj - EV$	D	5	3 hrs

Table 3 Complexity and Run Times

For the CCP and EV models, multiple iterations are required to determine the appropriate α and Γ values, respectively. Because there is no prior indication if a particular protection α_i or Γ_i (for flight i) renders the model infeasible, experimentation with different values is necessary. We found, for example, that for N_2, CCP was infeasible with α values of 99% and 95%. Also, some flights can be protected to a greater extent than others. To obtain the 'right' protection levels, the model had to be solved multiple times. We overcame this limitation using our $\alpha - CCP$, $\Delta - EV$ and $\Delta - Obj - EV$ models. Although they each had to be solved only once, the computation time of $\alpha - CCP$, $\Delta - EV$ and $\Delta - Obj - EV$ models are comparable to the time required to solve a single iteration of CCP, EV and $Obj - EV$ respectively.

	% Flight Delays (min)					Pax Disruptions	
	≤ 15	≤ 60	≤ 90	≤ 120	≤ 180	#D-pax	D-pax reduced
$\alpha_i = 90 \, \forall \, i$	78.54	93.10	95.63	97.82	98.91	1025	6.77
$\alpha_i = 92 \, \forall \, i$	77.54	92.54	95.00	97.36	98.54	1209	-9.9
$\alpha_i = 94 \, \forall \, i$	79.54	93.73	96.00	98.18	99.18	987	10.2
Airline's Routing	77.72	92.82	95.3	97.73	98.91	1100	0.00

Table 4 Robustness metrics for N_2 do not improve with increasing protection parameters in the CCP model (t = 90)

4.2. Correlations between protection levels and robustness metrics

The protection parameters α and Γ in the CCP, EV and $Obj-EV$ models are designed to represent the extent of robustness desired, with larger values of α and Γ representing higher levels of solution robustness and improved robustness metrics values. Notice, from Table 4, that for individual flight protection levels α_i for flight i for a delay threshold $t = 90$, we get a network on-time performance for $t = 90$ minutes, of at least $\alpha = \alpha_i$ for all i. We observe, however, that the values of the robustness metrics do not necessarily increase for solutions to the EV, $Obj-EV$ and CCP models using increased values of protection parameters.

Table 4 and Figure 4 show that with increases in α, the solutions to CCP can worsen with respect to flight on-time performance, passenger disruptions and total aircraft delay minutes. There are several explanations for this. First, optimal solutions to the CCP with $\alpha_i = 90$ for all $i \in F$ can include solutions that satisfy $\alpha_i = 90$ or $\alpha_i = 94$ or $\alpha_i = 96$ for all $i \in F$. All of these solutions are considered optimal although, intuitively, the solution with the highest α value has the most slack and should therefore be the most 'robust'. As a result, non-monotonicity of robustness metrics occurs, as shown in Table 4. In the case of $\alpha_i = 90$ for all $i \in F$, 95.63% of the flights are delayed less than t=90 minutes and for $\alpha_i = 92$ for all $i \in F$, 95% of the flights are delayed less than t=90 minutes.

Another reason that the robustness metric values do not always increase with increasing values of α is that the CCP model focuses on selecting routings that limit the likelihood of occurrence of 'long' flight delays as a proxy for robustness, and is not (and cannot) be formulated such that its solution will optimize simultaneously the three different (and sometimes opposing) robustness metrics that we evaluate.

Similarly, as illustrated in Figure 5 and Table 5, higher values of Γ in EV do not always produce better solutions. The explanations for this occurrence are similar to that for CCP. Multiple optimal solutions to EV and $Obj-EV$ for given Γ-values satisfy protection values Γ, even though the solutions might have very different levels of slack and hence, exhibit very different performance with respect to our robustness metrics. In fact, some of the optimal EV solutions are less robust

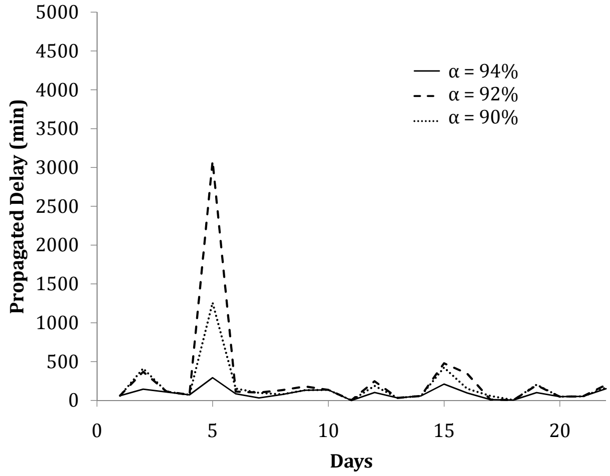


Figure 4 CCP **model solutions for network** N_2 **do not show improved total delay minutes with increased protection (**t **= 90)**

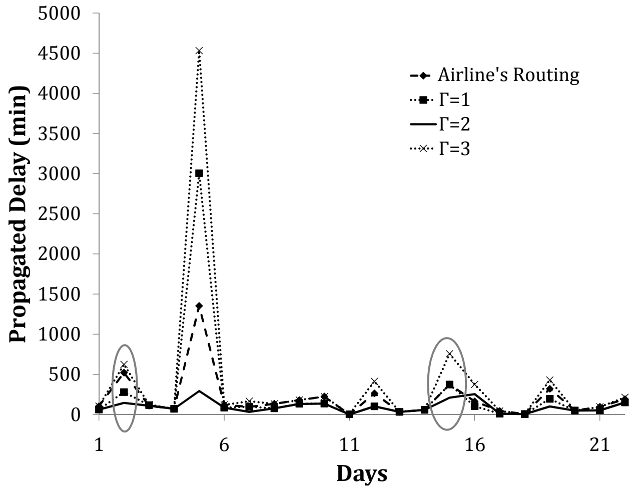


Figure 5 **Bertsimas-Sim model solutions for** N_2 **show non-monotonic relationship of propagated delay with** Γ **(**t **= 90)**

with respect to our robustness metrics than the airline's routing, as is the case with $\Gamma = 3$ in Figure 5 and Table 5. The second explanation for why higher levels of Γ in EV and $Obj-EV$ models can lead to less robust solutions is the same as that for CCP. EV, like CCP, does not capture all the robustness metrics precisely, but rather builds robustness into the solution by over-covering flight legs with multiple aircraft if some have associated delays that can be long in the extreme-case. Similar is the case with $Obj-EV$. We, however, see slightly less variability in the alternate optima because the $Obj-EV$ model is based on extreme propagated delay minutes for each string and accounts for every minute of delay, instead of thresholds of t minutes.

The issue of choosing solutions with the maximum α and Γ values among multiple optimal solutions is addressed by the $\alpha-CCP$, $\Delta-EV$ and $\Delta-Obj-EV$ respectively, with their objective

	Flight Delays				Pax Disruptions	
	$\leq$ 15 min	$\leq$ 60 min	$\leq$ 120 min	$\leq$ 180 min	#D-pax	D-pax reduced
$\Gamma_i = 1\,\forall\, i$	78.63	93	97.36	98.54	1222	-11
$\Gamma_i = 2\,\forall\, i$	79.54	93.54	98.09	99.18	987	10.3
$\Gamma_i = 3\,\forall\, i$	74.18	91	96.54	97.90	1353	-23
Airline's Routing	77.72	92.82	97.73	98.91	1100	0

Table 5 **Non-monotonicity in robustness metrics for N_2 with increase in Γ in EV (t = 90)**

	Flight Delays				Pax Disruptions	
	$\leq$ 15 min	$\leq$ 60 min	$\leq$ 120 min	$\leq$ 180 min	#D-pax	D-pax reduced
$\Delta - EV$	79.54	93.54	98.09	99.18	987	10.3
$\Delta - EV$ (alternate)	78.2	93.1	97.73	98.82	1056	4.03
$\Delta - Obj - EV$	79.27	93.54	98.1	99.1	988	10.14
$\Delta - Obj - EV$ (alternate)	78.36	93.1	97.9	99	1012	7.93
$\alpha - CCP$	79.54	93.73	98.18	99.18	987	10.2
LCB	79.54	93.73	98.2	99.18	986	10.3
Airline's Routing	77.72	92.82	97.73	98.91	1100	0

Table 6 **$\Delta - EV$, $\Delta - Obj - EV$ and $\alpha - CCP$ identify robustness parameters to improve upon the airline's routing for N_2 (t = 90)**

functions to maximize the protection levels. Table 6 shows that these models can select, among different optimal solutions to the CCP and EV models, those with the highest levels of protection parameters, and it turns out, the greatest values of our robustness metrics.

4.3. Solution Differences due to Modeling Paradigms

In this section, we compare the three modeling paradigms: extreme-value based, probabilistic chance-constrained-based and the tailored approach studied in this paper. To avoid issues with specification of robustness parameters and to focus on the modeling paradigms, we compare the LCB, $\alpha - CCP$, $\Delta - EV$ and $\Delta - Obj - EV$ models.

Table 6 compares the airline's routing with solutions obtained from the $\Delta - EV$, $\Delta - Obj - EV$, $\alpha - CCP$ and LCB models. First, both the $\Delta - EV$ and $\alpha - CCP$ model solutions improve upon the airline's routing, overcoming the drawback with some of the solutions generated by the EV and CCP models. In fact, $\Delta - EV$, $\Delta - Obj - EV$, $\alpha - CCP$ and LCB perform similarly with respect to our metrics. The improvements in the 15 minute on-time performance results, for this data, in the airline's ranking improving to place second among US carriers. The delay minutes saved result in a savings of $120,000 for the 22 days, when the per-minute costs are according to Air Transport Association (2009). Second, the $\Delta - EV$ model, like EV, can still have multiple optimal solutions. Although alternative optimal solutions have the same Δ-value, they differ significantly in the associated values of the robustness metrics.

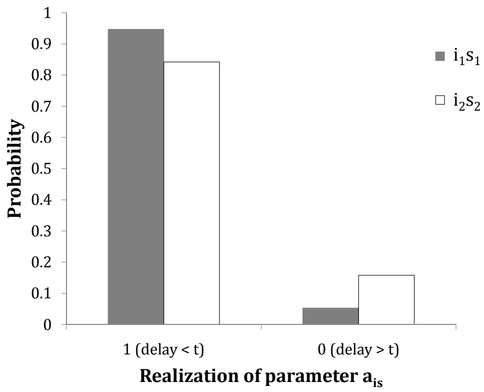
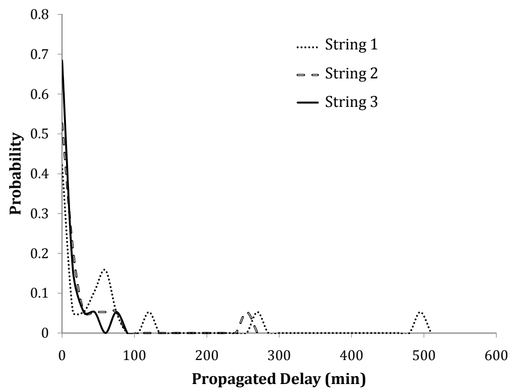


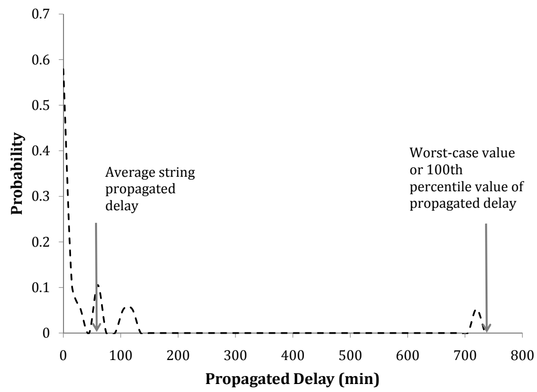
Figure 6 a_{is} **realization probabilities,** N_2

Consider $\Delta - EV$ and $\Delta - Obj - EV$ solutions in Table 6. We see that the $\Delta - EV$ and $\Delta - Obj - EV$ solution performs similar to the LCB and $\alpha - CCP$ models; however their alternate optima $\Delta - EV$ (alternate) and $\Delta - Obj - EV$(alternate) perform significantly differently. In fact, $\Delta - EV$ and $\Delta - EV$(alternate) both ensure that of 44 of 50 flights in N_2 are covered (have delays less than t) in the extreme case, but have very different *probabilities* of flight delay less than $t = 90$, of 91% and 87% respectively. This results in a total delay difference of 2091 minutes over 22 days between the two solutions. A similar difference is observed between $\Delta - Obj - EV$ and $\Delta - Obj - EV$(alternate). These differences are much lower, however, in the case of worst-case delay metrics (such as delays more than 180 minutes), because these models are driven by the extreme value delays. Because the $\Delta - EV$ and $\Delta - Obj - EV$ models are formulated to avoid low probability values of worst-case delay or non-coverage, optimal solutions can have very differing values of average performance as measured by on-time performance and number of disrupted passengers averaged over several days. To illustrate, consider Figure 6 in which realization probabilities of two flight-string combinations i_1s_1 and i_2s_2 are shown. $\Delta - EV$ and EV do not distinguish between both these realizations, because both i_1s_1 and i_2s_2 have non-zero historical probability of realizing a value of 0. Also consider Figure 7(b) in which the distributions of propagated delays for strings in N_1 are shown. The propagated delay distribution is seen to be roughly bi-modal, with a large (≈88%-95%) probability of the propagated string delay being on the lower end of the scale, and a small (≈5%-12%) probability of the propagated delay being close to its worst-case value. The probability of interim values occurring is very small. Due to this reason, the emphasis on extreme values of delays does not necessarily drive the extreme-value models towards good values of the average-case robustness metrics.

Because the $\alpha - CCP$ model captures the probabilistic nature of events, its emphasis on high-probability events results in optimal solutions which correlate on average to improved values of



(a)String Propagated Delays



(b)Average and Extreme Values of Propagated Delays of Strings

Figure 7 Propagated Delay Distributions of Strings

the metrics under consideration (Table 6). Alternative optima to $\alpha - CCP$ exist, but there is little difference in their performance with respect to the values of robustness metrics.

The LCB model uses average values of string propagated delay, shown in Fig. 7(b). Because the average is at about the 85th or 90th percentile of string propagated delay, LCB ignores the extreme value occurrences forming the remaining 5-10% of the distribution, and its objective of minimizing total average propagated delay seems to correlate well with our robustness metrics, with the LCB solutions performing well for all metrics (Table 6). Although the LCB model does not capture the probability distribution of delays, by using average delay values, it is a special case of the string-based $\alpha - CCP$ model (when the protection level in the $\alpha - CCP$ model is set, for these problems, to the 90th percentile level). The advantage of LCB over $\alpha - CCP$ is that it is able to capture uncertainty in a simpler way, resulting in a highly tractable model.

We can apply the insights gained in solving the aircraft routing problem with LCB and $\alpha - CCP$ models to improve the extreme value-based robust optimization models. Specifically, by adjusting the 'extreme' values in the $Obj - EV$ and $\Delta - Obj - EV$ models to, say, the average values of string propagated delay, we can generate solutions using these models that perform more like those of the LCB and $\alpha - CCP$ models. This, of course, underscores the difficulty of setting these extreme values a priori, and the sensitivity of the solutions to these model inputs.

5. Conclusions

In this paper, we study the application of three types of models - extreme-value based, probabilistic and tailored approaches, to the problem of aircraft routing. These three robustness mechanisms lead to different models with different solutions which have different robustness performances with respect to various metrics of interest.

The extreme-value models (EV and $Obj-EV$) were based on the Bertsimas and Sim robust optimization approach, the probabilistic model (CCP) on Charnes and Cooper's chance-constrained programming approach and the tailored model (LCB) was Lan, Clarke and Barnhart's robust aircraft routing approach. Increased complexity and solution times are associated with the extreme value and probabilistic models, when compared to a deterministic model, because the models have to be solved several times for different values of the robustness parameters Γ and α. To avoid iterative re-solving, we developed extensions to these models: $\Delta - EV$, $\Delta - Obj - EV$ for the extreme-value approach, and $\alpha - CCP$ for the probabilistic approaches respectively. Our extended models can be solved in a single iteration, with runtimes equivalent or lower than a single iteration of the basic models. We evaluated solutions to the different models through simulation, and measure performance via total aircraft delay, on-time performance metrics, and passenger disruption metrics. The extended extreme value and probabilistic models can consistently lead to the generation of more robust solutions (compared to the basic models, and the solution currently operated by the airline), as defined by the metrics of interest. The more general versions of our extended models, applicable to the broad class of problems involving network-based resource allocation, are presented in Appendices B and C.

Our findings show that extreme-value based models EV, $Obj - EV$, and $\Delta - EV$ and $\Delta - Obj - EV$ have optimal alternative solutions that exhibit very different performances according to our robustness metrics; varying from good improvement compared to the airline's routing, to no significant improvement or even deterioration. This behavior is because the robustness mechanism (Γ or Δ) is driven by *extreme* values of delay. This dependence on extreme delay values, ignoring probabilistic information, leads in some cases to a large disparity in the performance of the alternative optimal solutions. In such cases, extra care should be taken in evaluating alternative optimal solutions to these approaches. From this, we conclude that it is not effective to drive the solution process with extreme values that are rare.

The tailored approach LCB and the probabilistic CCP robustness approaches are very similar, in that expected values of string propagated delay, used to drive the LCB, are at about the 85th to 90th percentile of the string propagated delay distribution; and cause the formulation to focus on higher-probability events. Similarly, probabilistic approaches also focus on higher-probability delay events, and produce improved routings according to our metrics. These approaches thus capture more information about the system and focus on more likely delay events, and thus are more in line with our metrics of interest. Though the tailored approach in itself does not explicitly capture knowledge of probability distributions, by simplistically incorporating the 'right' delay quantile in its objective, it can achieve improved results through a less complex model. The probabilistic approaches (CCP, $\alpha - CCP$) allow more fine-tuning of robustness using the α and t parameters, but at a cost of a larger, more complex formulation (albeit very tractable for this application) and the implementation cost that additional distribution knowledge is needed for the $\alpha - CCP$ compared to LCB.

In conclusion, the efficacy of any given robust approach is determined not by the approach or model alone, but by the interaction between the model, data and evaluation metrics. Our paper underscores the importance of choosing an approach that aligns itself well with the data distributions for the aircraft routing problem and metrics of interest to the DoT, industry and passengers. When applying general robust approaches to more specific problems, care should be taken to understand the nature of uncertainty and in choosing robustness parameters, in relation to the metrics, especially when metrics involving multiple stakeholders are involved.

Appendix A: $Obj - EV$ model Formulation

The $Obj - EV$ model plans for the case when a user-defined number of strings Γ in the formulation realize their *worst-case values* of propagated delay, while other strings do not vary from the assumed deterministic or nominal values, namely, 0.

The protection parameter Γ is defined in the interval $[0, |S|]$. Let $\hat{d}_s$ be the worst-case (maximum or 100th percentile) value of propagated delay of string s observed in the historical data. (Note that this is not the worst possible realization of propagated delay, but only that in the selected period of historical data.)

The formulation minimizes the extreme (or worst-case) value of propagated delay caused by *any* set of Γ strings in the solution realizing their worst-case values, and the other strings attaining their nominal values of propagated delay (equal to zero). That is, it minimizes the propagated delay caused by the maximal subset of Γ strings in the solution realizing their extreme propagated delay values, as shown in (A.1). Variable w_s for string $s \in S$ takes on value 1 if string s is present in the solution, and the model plans for its extreme value being realized. The extreme-value formulation, according to Bertsimas and Sim (Bertsimas and Sim 2003), is as follows.

$$
\begin{aligned}
&\mathbf{Obj-EV}: \\
&\min \left\{ \sum_{s \in S} 0 x_s + \max_{\{S' \cup \{t\} | S' \subseteq S, |S'| = \lfloor \Gamma \rfloor, t \in S \setminus S'\}} \{ \sum_{s \in S'} \hat{d}_s w_s + (\Gamma - \lfloor \Gamma \rfloor) \hat{d}_t w_t \} \right\} && \text{(A.1)} \\
&\text{s.t. Cover, Balance, Count, and Integrality } (2.2) - (2.7) && \text{(A.2)} \\
&\quad x_s \leq w_s \qquad \forall\, s \in S && \text{(A.3)} \\
&\quad w_s \geq 0 \qquad \forall\, s \in S && \text{(A.4)}
\end{aligned}
$$

$Obj - EV$ can be linearized, and cast as a mixed integer program, as follows:

$$
\begin{aligned}
&\mathbf{Obj-EV}: \\
&\min z\Gamma + \sum_{s \in S} p_s && \text{(A.5)} \\
&\text{s.t. } z + p_s \geq \hat{d}_s w_s \qquad \forall\, s \in S && \text{(A.6)} \\
&\quad \text{Cover, Balance, Count, and Integrality } (2.2) - (2.7) && \text{(A.7)} \\
&\quad x_s \leq w_s \qquad \forall\, s \in S && \text{(A.8)} \\
&\quad w_s \geq 0 \qquad \forall\, s \in S && \text{(A.9)} \\
&\quad z \geq 0. && \text{(A.10)}
\end{aligned}
$$

These mixed integer programs have a very different structure from AR (Marla 2007), and face computational challenges in solving (§4.1).

Appendix B: Delta Model - General Formulation

The Delta model is designed to address the basic practical issue encountered in the Bertsimas and Sim approach; that of selecting an appropriate protection parameter Γ_i for each constraint i. This is a potentially cumbersome task for large-scale problems. Given this, it might be necessary to solve the Bertsimas-Sim robust optimization model repeatedly for varying values of the Γ_i parameters before a satisfactory solution is identified. In the case of large-scale network-based resource allocation problems, solving the model even once can be computationally challenging; and therefore the requirement to solve it multiple times is likely to be impractical for large problems. Network-based resource allocation problems are often formulated as binary integer programs, and our Delta model is particularly designed for such formulations.

The standard binary integer program that is required to be made robust is:

$$\max \quad \sum_{j \in J} c_j x_j \tag{B.1}$$

$$\text{s.t.} \sum_{j \in J} a_{ij} x_j \leq b_i \;\; \forall\, i \in I \tag{B.2}$$

$$x_j \in \{0,1\} \quad \forall\, j \in J \tag{B.3}$$

In (B.1)-(B.3), we use the following notation. I is the set of constraints, and J the set of variables, c_j is the profit coefficient for variable j and b_i is the right-hand side value for ith constraint, $\forall\, i \in I$. a_{ij} for all $i \in I$, $j \in J$ is the coefficient of variable j in constraint i. a_{ij} for all $i \in I$, $j \in \bar{J}_i$, is subject to uncertainty, with $\tilde{a}_{ij}$ its realized value. a_{ij} is the nominal value of $\tilde{a}_{ij}$, and also the mean value of its symmetric range of variation. $\hat{a}_{ij}$ is the half-interval of the symmetric range of variation of $\tilde{a}_{ij}$, for all $i \in I, j \in J$. $\hat{a}_{ij} = 0$ for $j \in J \backslash \bar{J}_i$. x_j for all $j \in J$, is a binary decision variable that equals 1 if variable is present in the solution and 0 otherwise.

To avoid the need to specify Γ values, we modify the Bertsimas-Sim formulation to include a constraint requiring the total profit of the robust solution to be within a difference of δ from the nominal optimal value. Additionally we change the objective to one of minimizing the maximum number of variables that *must* assume their nominal, rather than extreme, values to satisfy all constraints. We define *variable* Δ_i equal to the maximum number of variables x in the solution with $x = 1$ whose coefficient values are subject to uncertainty and must assume their nominal values for constraint i to remain feasible. We sort, for each constraint i, its associated columns $j \in \bar{J}_i$, in increasing order of their $\hat{a}_{ij}$ values (ties are broken arbitrarily). After ordering, the rank of the jth column in the ith constraint is denoted by $l(i,j)$. Also, the original index (j) of the variable that takes the lth position in the *sorted* $\hat{a}_{ij}$ values for constraint i is denoted by $j(i,l)$. For example, the variable j in constraint i with the smallest $\hat{a}_{ij}$ value has $l(i,j) = 1$. The variable with the largest $\hat{a}_{ij}$ value has $l(i,j) = N$, with N equal to the number of binary variables in J.

Let y_j^* be the optimal value of variable j for all $j \in J$ for the nominal problem (B.1)-(B.3). Then δ is the user-specified incremental cost that is acceptable for increased robustness, that is, the profit of a robust solution from the Delta formulation is at least $\sum_{j \in J} c_j y_j^* - \delta$. Let variables v_{ij} equal 1 if the uncertain coefficient $\tilde{a}_{ij}$ is not allowed to take on its extreme value, and takes on its nominal value in the solution of the Delta model. Variables w_{il} equal 1 for all $l \geq |J| - |\bar{J}_i| + 1$ in constraint i for which there exists a $k \geq l$ with $v_{ik} = 1$, for $l = |J| - |\bar{J}_i| + 1, ..., N+1$. w_{il}s for $l = |J| - |\bar{J}_i| + 1, ..., N+1$ in each constraint i follow a step function.

This leads to the following Delta formulation:

$$\min \sum_{i \in I} \Delta_i \tag{B.4}$$

$$\text{s.t.} \sum_{j \in J} c_j x_j \leq \sum_{j \in J} c_j y_j^* + \delta \tag{B.5}$$

$$\sum_{j \in J} (a_{ij} + \hat{a}_{ij}) x_j - \sum_{j \in J} \hat{a}_{ij} v_{ij} \leq b_i \qquad \forall i \in I \tag{B.6}$$

$$\Delta_i \geq \sum_{l=|J|-|J_i|+1}^{|J|} [(l - |J| + |J_i|)(w_{i,l} - w_{i,l+1}) + v_{i,j(i,l)} - w_{i,l}] \qquad \forall i \in I \tag{B.7}$$

$$\Delta_i \geq \sum_{j \in J} v_{ij} \qquad \forall i \in I \tag{B.8}$$

$$v_{ij} \leq x_j \qquad \forall j \in J, \forall i \in I \tag{B.9}$$

$$v_{i,j(i,l)} \leq w_{i,l} \qquad \forall l \in J \tag{B.10}$$

$$v_{ij} \geq x_j + w_{i,l(i,j)} - 1 \qquad \forall j \in J, \forall i \in I \tag{B.11}$$

$$w_{i,l+1} \leq w_{i,l} \qquad \forall l \in |J| - |\bar{J}_i| + 1, ..., |J|, \forall i \in I \tag{B.12}$$

$$w_{i,|J|-|J_i|+1} \leq 1 \qquad \forall i \in I \tag{B.13}$$

$$w_{i,|J|+1} = 0 \qquad \forall i \in I \tag{B.14}$$

$$w_{il} = 0 \qquad \forall l \in 1, ..., |J| - |\bar{J}_i|, \forall i \in I \tag{B.15}$$

$$x_j \in \{0, 1\} \qquad \forall j \in J \tag{B.16}$$

$$v_{ij} \in [0, 1] \qquad \forall j \in 1, ..., |J|, \forall i \in I \tag{B.17}$$

$$w_{il} \in \{0, 1\} \qquad \forall l \in 1, ..., |J| + 1, \forall i \in I \tag{B.18}$$

The formulation is described as follows: The objective (B.4) is to minimize the sum of maximum number of coefficients in each constraint, that must assume their nominal values to satisfy all constraints. This serves the purpose of trying to maximize the number of coefficients that can take their extreme values and still maintain feasibility. Constraints (B.5) require that the profit from the 'robust' solution not differ from the profit of the deterministic solution by more than a 'robustness budget' of δ. Feasibility is assured by constraints (B.6). We set v_{ij} equal to 1 in constraint i, for all coefficients j that *must* be set to their nominal values. Inequalities (B.9) prevent a coefficient from not taking on its extreme value unless the variable is present in the solution, that is, s_{ij} to zero if x_j is zero in the solution. Constraints (B.7) and (B.8) provide different mechanisms to count the maximum number of variables whose coefficients in constraint $i \in I$ whose coefficients are subject to uncertainty and must take on nominal values. The explanation for this constraint lies in the realization that when the columns are sorted in increasing order of $\hat{a}_{ij}$ values for each row i, the *maximum number* of coefficients that must assume nominal values to maintain feasibility is determined by forcing the smallest $\hat{a}_{ij}$s ($\hat{a}_{ij} \neq 0$) in the solution to have their associated v_{ij} values set to 1, if x_j is in the solution. Though constraints (B.7) and (B.8) are essentially the same, we retain both in order to provide better bounds on the solution. (B.10) requires the w variable to be at least as large as the corresponding v variable for that j and i. Also, (B.11) forces v to be 1 if both the corresponding x and w are 1. Constraints

(B.12), (B.13) and (B.14) require the w_{il} variables corresponding to uncertain coefficients in each i to form a step function. The w_{il} variables corresponding to coefficients that are not subject to uncertainty are zero (B.15). The x and w variables are binary, as required by (B.16) and (B.18) respectively. They thus force the v variables to take on binary values as well. Alternatively, one can think of this model as maximizing the minimum number of coefficients, summed over all constraints, that can take on their worst-case values, under budget limitations.

Alternative objective functions to this model include 1) minimizing a weighted sum of coefficients not allowed to realize their extreme values $\left(\sum_{i\in I} w_i\Delta_i\right)$ with weight i assigned to constraint $i \in I$; or 2) minimizing the maximum number of uncertain coefficients in any constraint that must assume nominal values rather than their extreme values, to satisfy the constraints; that is $\min \nu$, with additional constraints $\nu \geq \Delta_i$, for all $i \in I$.

Appendix C: Extended Chance-Constrained Programming - General Formulation

The Chance-Constrained Programming or CCP approach also faces the problem of specifying a probability of satisfaction for each constraint. This is potentially a limitation of the approach when applying to large problems. The Extended Chance-Constrained Programming ($ECCP$) approach avoids the need to specify the protection level for each constraint explicitly.

The standard linear program that is required to be made robust is:

$$\max \quad \mathbf{c}^T\mathbf{x} \tag{C.1}$$

$$\text{s.t.} \sum_{j\in J} a_{ij}x_j \leq b_i \;\; \forall\, i \in I \tag{C.2}$$

$$\mathbf{x} \geq 0, \tag{C.3}$$

where the notation used is the same as for (B.1)-(B.3). Its general chance-constrained formulation is as follows:

$$\max \quad f(\mathbf{c},\mathbf{x}) \tag{C.4}$$

$$\text{s.t.} \; P(\sum_{j\in J} a_{ij}x_j \leq b_i) \geq \alpha_i \;\; \forall\, i \in I \tag{C.5}$$

$$\mathbf{x} \in \mathbf{X}, \tag{C.6}$$

where ‘P’ means ‘probability’, and $\mathbf{X}$ is the feasible set for (C.1)-(C.3). α_i ($0 \leq \alpha_i \leq 1$) for all i, is a user-specified *protection level* for constraint i, and (1 - α_i) specifies the maximum degree of violation of constraint i.

Charnes and Cooper translate (C.4)-(C.6) into different models, for varied types of objective functions and correspondingly different constraints. They present models for three types of objective functions, further details of which are available in (Charnes and Cooper 1963), (Charnes and Kirby November 1967). In each of these models, **b** and **c** are assumed to be uncertain. We retain these assumptions in our model.

Ben-Israel (Ben-Israel 1962) shows that the chance-constraint (C.5) can be linearized as follows:

$$P(\sum_{j=1}^{N} a_{ij}x_j \leq b_i) \geq \alpha_i \Leftrightarrow \sum_{j=1}^{N} a_{ij}x_j \leq F_{b_i}^{-1}(1-\alpha_i), \tag{C.7}$$

with $y = F_{b_i}^{-1}(1-\alpha_i)$ equal to the *quantile* value in the cumulative distribution function (CDF), F_{b_i}, of b_i such that the probability that b_i takes on values less than or equal to y is (1- $\boldsymbol{\alpha}$). That is, if $f(b_i)$ is the probability distribution function of b_i, $\int_{-\infty}^{y} f(b_i)db_i = 1-\alpha_i$.

Further, from the distributions of the elements in $\mathbf{c}$, we can determine a vector of stipulations β such that $P(\mathbf{c} \leq \lambda_c) \geq \beta$ for some λ. Ben-Israel shows that we can also write the above linear program as

$$\max (F_{\mathbf{c}}^{-1}(\beta))^T \mathbf{x} \tag{C.8}$$

$$\text{s.t.} \mathbf{A}\mathbf{x} \leq F_{\mathbf{b}}^{-1}(1-\alpha) \tag{C.9}$$

$$\mathbf{x} \geq 0, \tag{C.10}$$

In our extended model, to avoid the need to specify the protection level for each constraint explicitly, we include a constraint on the overall expected profit of the robust solution and change the objective to one of maximizing the sum total of protection level provided for all constraints, as follows:

$$\max \sum_{i \in I} \alpha_i \tag{C.11}$$

$$\text{s.t.} P(\mathbf{A}\mathbf{x} \leq \mathbf{b}) \geq \alpha \tag{C.12}$$

$$E(\mathbf{c}^T\mathbf{x}) \geq \mathbf{c}^T\mathbf{y}^* - \delta \tag{C.13}$$

$$\mathbf{x} \geq 0 \tag{C.14}$$

$$\alpha \geq 0, \tag{C.15}$$

where $\mathbf{c}^T\mathbf{y}^*$ is the expected profit of the nominal optimal solution $\mathbf{y}^*$ to (C.1)-(C.3). Alternative objective functions may include 1) maximizing the minimum protection level α_{min} with an added constraint $\alpha_{min} \leq \alpha_i$ for all $i \in I$; or 2) maximizing a weighted sum of the constraint protection levels ($\sum_{i \in I} w_i\gamma_i$), w_i being the non-negative weight assigned to constraint $i \in I$.

To linearize (C.11) - (C.15), we require the knowledge of some quantiles and their associated values of the probability distribution of the right-hand-side b_i, for each constraint i, and the expected values of the profit function. Let b_i is the nominal value for the right-hand-side of the ith constraint, and c_j the expected profit coefficient corresponding to the jth variable, for all $j \in J$;. Let K_i represent the set of discretized protection levels known for b_i. We now set the protection levels α_i for each constraint i as variables, representing the quantile that is chosen from among the K_i available quantiles. Let b_i^k be the kth quantile value of the RHS parameter of the ith constraint, for all $k \in K_i, i \in I$; and z_j^* be the optimal solution to (C.1) - (C.3) found using nominal values of the $\mathbf{b}$ and $\mathbf{c}$ parameters, for all $j \in J$. p_i^k is the protection level probability associated with quantile $k \in K$ for constraint $i \in I$. The objective function value, denoted α_{min}, equals the minimum protection level achieved over all constraints $i \in I$. To capture the tradeoff of robustness with profits, we assume that the planner is willing to forego a (user-specified) profit of δ to instead gain a robust plan.

Decision variables y_i^k are binary variables that equal 1 if the protection level (expressed as a probability p_i^k, with $0 \leq p_i^k \leq 1$) represented by the kth quantile ($k \in K_i$) is attained in constraint $i \in I$; and 0 otherwise. This means that if the kth quantile value is protected against, the $(k+1)$st quantile is also automatically protected against. This follows from the fact that constraints are "less than" inequalities. The y_i^k values for any constraint i, follow a step function. Variables α_i represent the protection level attained for the ith constraint, for all $i \in I$.

The extended chance-constrained model (ECCP) is as follows:

$$\max \sum_{i \in I} \alpha_i \tag{C.16}$$

$$\text{s.t.} \sum_{j \in J} c_j x_j \geq \sum_{j \in J} c_j z_j^* - \Delta \tag{C.17}$$

$$\sum_{j \in J} a_{ij} x_j \leq \sum_{k=1}^{K} b_i^k (y_i^k - y_i^{k-1}) \qquad \forall\, i \in I \tag{C.18}$$

$$y_i^k \geq y_i^{k-1} \qquad \forall\, k = 1, ..., K_i, i \in I, \tag{C.19}$$

$$y_i^0 = 0 \qquad \forall\, i \in I \tag{C.20}$$

$$y_i^K = 1 \qquad \forall\, i \in I \tag{C.21}$$

$$\alpha_i \leq \sum_{k=1}^{K_i} p_i^k (y_i^k - y_i^{k-1}) \qquad \forall\, i \in I \tag{C.22}$$

$$x_j \geq 0 \qquad \forall\, j \in J \tag{C.23}$$

$$y_i^k \in \{0, 1\} \qquad \forall\, k \in K_i, i \in I \tag{C.24}$$

$$0 \leq \gamma_i \leq 1 \qquad \forall\, i \in I \tag{C.25}$$

The objective function (C.16) maximizes the total probability that each constraint $i \in I$ is feasible. It can also be re-written to maximize the minimum value of γ_i over all constraints. (C.17) ensures that the solution's expected profit is within Δ units of the expected profit associated with the nominal optimal solution (found by solving the problem using nominal values of the **b** vector). For all constraints $i \in I$, (C.18) forces the left-hand-side (LHS) to be less than or equal to b_i^k if y_i^k equals 1, thereby ensuring constraint satisfaction with at least the probability associated with quantile k. For the smallest quantile k^* that can be satisfied, the $y_i^{k^*}$ value is 1, and quantiles $k < k^*$ have $y_i^k = 0$. Thus, the RHS value of this constraint is selected as the smallest one that can be satisfied by the solution. (C.19) ensures that the y_i^ks are monotonically increasing and follow a step function, such that if a smaller quantile (higher protection) is achieved, the larger quantile (lower protection) is automatically achieved. (C.20) and (C.21) set the boundary values of the y_i^k step functions. Constraints (C.22) set γ_i to be no greater than the highest protection level provided to constraint i by the solution. The x_js are non-negative for all $j \in J$; y_i^ks are binary for all $j \in J$ and all $k \in K_i$ for all $i \in I$; and γ_is are non-negative for all $i \in I$ as required by (C.23),(C.24) and (C.25), respectively.

References

Air Traffic Association. 2008. Annual Report. http://www.iata.org/NR/rdonlyres/84158349-7772-4892-86AB-836DE73E0A52/0/IATAAnnualReport2008.pdf.

Air Transport Association. 2009. Ata annual and per-minute cost of delays to u.s. airlines. http://www.airlines.org/Economics/DataAnalysis/Pages/CostofDelays.aspx.

Armacost, A., C. Barnhart, K. Ware. 2002. Composite variable formulations for express shipment service network design. *Transportation Science* **36** 1–20.

Ben-Israel, A. 1962. On some problems of mathematical programming. Ph.D. thesis, Northwestern University.

Bertsimas, D., M. Sim. 2003. Robust discrete optimization and network flows. *Mathematical Programming Series B* **98** 49–71.

Bertsimas, D., M. Sim. 2004. The price of robustness. *Operations Research* **52(1)** 35–53.

Bureau of Transportation Statistics. 2009a. http://www.transtats.bts.gov/HomeDrillChart.asp.

Bureau of Transportation Statistics. 2009b. Airline Service Quality Performance Database. http://aspm.faa.gov/asqp/sys/.

Bureau of Transportation Statistics. 2009c. On-time Performance - Flight Delays at a Glance. http://www.transtats.bts.gov/HomeDrillChart.asp.

Charnes, A., W. W. Cooper. 1959. Chance constrained programming. *Management Science* **6(1)** 73–79.

Charnes, A., W. W. Cooper. 1963. Deterministic equivalents for optimizing and satisficing under chance constraints. *Operations Research* **11(1)** 18–39.

Charnes, A., M. Kirby. November 1967. Some special p-models in chance-constrained programming. *Management Science* **14(3)** 183195.

CNNMoney.com. May 22, 2008. http://money.cnn.com/2008/05/22/news/economy/airline_delay/index.htm.

International Air Transport Association. 2007. Annual traffic growth data for year 2007 (Jan-Oct).

International Civil Aviation Organization. December 2007. Icao news release. http://www.icao.int/icao/en/nr/2007/pio200713_e.pdf.

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

Marla, L. 2007. Robust optimization for network-based resource allocation problems. Masters thesis, Massachusetts Institute of Technology.

US DoT. 2009. United States Department of Transportation website. http://www.dot.gov/new/index.htm.