How to evaluate the robustness of airlines schedules

Virot Chiraphadhanakul* and Niklaus Eggenberg†

October 6, 2009

Abstract

Airlines' schedules are built such as to maximize expected profit. Such schedules turn out to be more sensitive to delays and are hence unstable. The trend has thus evolved towards *robust* schedules, trading off between sensitivity with respect to disruptions (and hence lower delay costs) and higher operational costs in the deterministic schedule.

In this paper, we discuss the different ways to evaluate the robustness. We first show that the definition of robustness is not unique, and mainly differs in the way it is modeled and evaluated.

We compare different models for the Maintenance Routing Problem (MRP) according to the most common robustness metrics. We use data of a real airline to evaluate the robustness of different models aiming at increasing total slack in order to reduce delay propagation.

We show that some of the robustness metrics are correlated but not necessarily positively. Furthermore, we show that for a same metric, the efficiency varies depending on several factors such as the objective of the model, whether or not the model uses historical data and in which way. We show that no solution is globally better than the others, but that all of them improve the original schedule.

1 Introduction

The air transportation business is a highly developing market, the number of carried passengers being continuously increased. The Federal Aviation Administration (FAA) estimates the number of flights to increase at a rate of 2.5% per year until 2025 (*FAA Aerospace Forecast Fiscal Years 2008-2025*, 2008). However, the profit margin for airlines is thin, especially with the latest fuel price increase. According to the International Air Transport Association (IATA) annual report 2008 (*Annual Report 2008*, 2008), the global airline profit reached $5.6 billion in 2007, which is less than a 2% margin on the total revenues of more than $490 billion.

In addition to such a thin profit margin, airlines experience recurrent delays, which incur extremely high costs to airlines themselves, but also to the transported passengers, which has a global impact on the whole economy. The Joint Economic Committee (JEC) estimates the delay costs of the US air traffic in 2007 at an alarming $41 billion (*Your Flight Has Been Delayed Again*, 2008): $19.1 billion for additional operating costs, with additional 740 million gallons of jet fuel consumed for delayed flights, $12 billion for value of passenger time and $9.6 billion spillover costs to the economy. The total flight arrival delay is estimated at 2.8 million hours, and JEC estimates that 20% of the total domestic flight time in 2007 was spent in delays.

Given the huge costs associated to delays and the small profit margin for airlines, it seems intuitive to focus on reducing delays, even if this implies some lost of expected revenue. This trade-off is known as the *cost of robustness* (Bertsimas and Sim, 2004): in the case of airline scheduling, this means a loss of expected revenue to decrease the delay costs. Achieving robust airline schedules is an active field in research.

*Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA

†TRANSP-OR laboratory, École Polytechnique Fédérale de Lausanne,CH-1015 Lausanne, Switzerland

The definition of robustness in the literature is, however, not unique: most often it stands for some stability metric of the solution according to varying data, i.e. the ability of a solution to remain feasible with respect to varying data. It may, however, also refer to solutions that are easier to recover in case the solution is unfeasible. Furthermore, even when the term is used with the same meaning, it is not clear how to determine whether one solution is more robust than another.

We identify four key points on which robustness depends, namely:

1. metrics;
2. models;
3. evaluation;
4. data.

Indeed, robustness is usually defined as a metric, which thus leads to many different types of robustness. Furthermore, even for a same metric, the way it is modeled and the data used by the model is also an important factor. Finally, the key point of determining the robustness is the way the solution is evaluated, which depends on both the performance metrics and the used data.

In this paper, we study the interactions between the four points defining robustness. We provide a case study on data from a real airline to show these interactions using different models to solve the Maintenance Routing Problem (MRP). We show that different metrics indeed lead to solutions with different properties. Furthermore, using both a priori and a posteriori evaluation of the solution, we show that some performance metrics are positively and some others negatively correlated, and that none of the solutions is globally better than the others. Finally, we compare different models on a same metric, one based on historical data to allocate slack where highest delays are expected and another that simply maximizes slack. We show that the using historical data can improve the solution to a larger extent than the myopic method, but that it depends on the way historical data is used.

The paper is structured as follows: section 2 reviews the studies on robust airline scheduling. Section 3 describes different models we use to compute robust solutions to the MRP. In section 4, we present a detailed case study for a real airline. Section 5 concludes this paper and proposes future research directions.

2 Evaluating robustness of airline schedules

The airline scheduling problem has been widely studied in the past decades. As the whole problem of airline scheduling is globally considered as intractable for large airlines, the scheduling approach is sequentially divided into different stages, each stage taking as input the solution(s) of the previous stages. The stages are the route choice problem, the fleet assignment problem, the maintenance routing problem (MRP), the crew pairing and finally the crew rostering problems. For general surveys on airline scheduling, see Weide (2009), Kohl et al. (2007) or Clausen et al. (forthcoming).

In the literature, there are two distinct approaches to evaluate the performance of robust schedules. The former method uses a qualitative estimation of robustness, looking at structural properties of the solution. The solution is, however, not tested on real/simulated scenarios. In the latter approach, the schedule is evaluated on a set of scenarios on which a recovery scheme is applied. A posteriori evaluation is thus based on observed metrics, whereas a priori evaluation relies on predictive metrics.

As discussed by Kohl et al. (2007), a recovery scheme must satisfy three objectives, namely (a) deliver the service to the passengers, (b) minimize the costs associated to the recovery and (c) recover the initial schedule as soon as possible. A robust schedule should have increased slack to absorb delays, exploit probabilities of flight delays and increase overlaps that potentially reduce recovery costs.

We divide the section according to the type of performance metrics used to evaluate the quality of a solution, starting with the a priori ones.

A priori evaluation. Ageeva (2000) consider the robust MRP, where robustness is defined as the number of *overlaps* of different aircraft routes. The robustness is purely a priori and schedules are compared on a priori values only.

Ehrgott and Ryan (2000) address the tour of duty problem for crews of Air New Zealand. Robustness is modeled by penalizing crew changing aircraft in a tour of duty. The authors only use this metric to evaluate the performance of the schedule.

Shebalov and Klabjan (2006) define robustness by *move-up crews*, i.e. by the number of possible crew swaps, which is a similar robustness metric than the number of overlaps for the MRP presented by Ageeva (2000).

Smith and Johnson (2006) use a similar definition of robustness, using the *station purity* to solve the robust fleet assignment problem. Station purity corresponds to plane on ground constraints at airport stations. The obtained solutions are compared with respect to estimated profits and maintenance costs.

Yen and Birge (2006) solve the crew pairing problem using a stochastic scheme using recourse. The performance limits on the obtained expected costs, including first-stage and recourse costs, on a limited number of scenarios. The authors show that the model achieves a significant gain on the value of expected cost, but do not recourse to simulation to test a posteriori performance.

A posteriori evaluation. For a posteriori evaluation, the schedule has to be adapted with respect to a certain disruption, which usually involves a recovery scheme. We therefore also review some studies on pure recovery to highlight the used a posteriori performance metrics.

Rosenberger et al. (2003a) present a stochastic model for the daily airlines' operations, resulting in the SimAir simulator which is used to evaluate different automated recovery policies. The used performance metrics are 15 and 60 minute on-time performance and the percentage of passenger misconnections (i.e. disrupted passengers). Rosenberger et al. (2004) extend the previous work by studying the robustness of fleet-assignments when using a short-cycle cancellation recovery algorithm. The simulations using SimAir restrict to aircraft routings only; a robust Fleet Assignment Models (FAM) with hub-isolation and and increased number of short cycles is proposed. The authors conclude that the resulting FAM are more robust than standard revenue maximization approaches, with respect to the metrics discussed in Rosenberger et al. (2003a) and also the number of aircraft swaps and the number of times the recovery scheme is called.

In his thesis, Bratu (2003) first discusses the different on-time performance metrics in the US. The conclusion is that the 15-miute on-time performance metric for aircraft is not a good predictor for passenger delay statistics. The author then presents the Passenger Delay Calculator, which reallocates passengers on a recovered schedule. Canceled passengers are assigned a fixed cost corresponding to the mean delay observed from historical data, multiplied by the average delay cost per minute.

Kang (2004) introduces the concept of *degradable* airline schedule, which consist of several independent sub-schedules or *layers*; the layer a flight belongs to is determined by its revenue through partitioning models. Performance is evaluated both a priori and a posteriori, using the MEANS simulator to generate disruptions. Performance is then estimated with respect to number of canceled flights, average flight delay, number of disrupted passengers, number of canceled passengers, average passenger delay and on-time probability for both aircraft and passenger.

Listes and Dekker (2005) present a stochastic scenario aggregation-based approach for robust fleet assignment. The authors discuss the validation of a solution, concluding that simulated a posteriori statistics are more concluding than a priori expectations on the scenarios used for optimization. Used performance metrics are load factor, spill percentage, total revenues, total operational costs, and total profit. The recovery strategy is to re-assign the fleet in a best

possible way using some additional side constraints such as initial plane location. The authors show that more robust solutions have indeed higher costs, but also higher profit.

Schaefer et al. (2005) address the crew scheduling problem under uncertainty, and derive two models to derive robust schedules. The first approach uses a penalty method, penalizing pairings with attributes close to the legal bounds, such as rest times, sit times or flight time. The second aims at minimizing estimation of the expected cost of a pairing. The expected cost is estimated using historical data over 50 to 500 days. The computational results compare the *fleet-time credit*, an evaluation of the difference between a duty's total cost and the total block time. The results show that the real crew cost is about 90% higher than the deterministically planned cost; the expected cost minimization schedule reduces by a few percent the differences.

Bratu and Barnhart (2006) present an embedded aircraft and crew recovery algorithm. The efficiency of a recovery scheme is evaluated according to 15-minute on-time performance, percentage of flights delayed by more than 45 minutes, percentage of delayed flights, number of canceled flights and average flight delay. Passenger statistics are total passenger delay, number of disrupted passengers, number of canceled passengers, and other passenger delay statistics such as average delay of disrupted passengers and average non-disrupted passenger delay. The results show that some of these metrics are inversely correlated. Note that passenger delay statistics are computed using the Passenger Delay Calculator of Bratu (2003).

Lan et al. (2006) present two different flight re-timing models minimizing delay propagation and the potential number of passenger disruptions, respectively. Both models are based on a delay distribution obtained from historical data. The models use the estimated expected delays to create the robust schedules. Evaluation for the MRP models is performed with respect to propagated delay and 15, 60 and 120 minutes on-time performance. For the passenger models, comparison is performed with respect to the number of disrupted passengers and the total passenger delay (delay statistics are computed using the Passenger Delay Calculator of Bratu, 2003). The delay propagation minimization model reduces propagated delay by 44%, and the number of disrupted passengers by 11%. The misconnection model reduces the total passenger delay by up to 20%, the number of disrupted passengers being reduced by about 40%.

AhmadBeygi et al. (2008) consider a flight retiming model minimizing the propagated delay while ensuring both routings and connections remain as in the original schedule. Simulations are performed using synthetic scenarios generated with the same probability distribution than the one used for the optimization model. The solutions are evaluated according to the value of the objective function of their models: *single-layer* or *multi-layer* models, which both account for delay propagation. The authors first evaluate performance on the deterministic scenario and then using simulated instances for which flight delays are generated and flights are pushed-back accordingly. Results show that propagated delay can be substantially reduced, the maximum delay propagation being at 50.9% in average for the generic scenarios.

Burke et al. (forthcoming) differentiate the *flexibility* and the *stability* (or *reliability*) of a schedule, the former being defined by the number of available recovery options, the latter according to the probability of a flight to be on-time. The authors show that reliability and flexibility, as they define it, are negatively correlated, i.e. more recovery options imply lower on-time probability for flights. Results are obtained by a specific simulator developed by KLM.

Lately, Eggenberg and Salani (2009) propose a general re-timing framework to increase both a schedule's robustness and its recoverability. The approach does however not consider lost connections due to retiming. Solutions are first evaluated a priori according to a priori structural properties such as total slack, minimum slack, average slack per aircraft and per flight and number of plane crossings. The solutions are then evaluated after application of a recovery algorithm. A posteriori performance is evaluated according to an external cost metric and number of canceled flights and passengers, total aircraft and passenger delay and number of rerouted passengers. The solutions performing best in average are obtained by maximizing the sum of each aircraft's minimum slack; in average, the resulting solutions save 56% of the recovery costs; the average number of lost passengers due to retiming for these solutions is 1.11%.

summary Clearly, the literature does not agree on the definition of robustness. We believe that robustness is an a priori concept whose efficiency should be evaluated both a priori and a posteriori metrics. We see that several models focus on specific metrics and that different metrics might be negatively correlated. Hence, as pointed out by Burke et al. (forthcoming), there are multiple non-dominated solutions, i.e. no absolute robust solution exists.

The focus of this paper is to study robustness for the MRP using both a priori and a posteriori metrics. To do so, we adopt a similar approach than AhmadBeygi et al. (2008), i.e. we compare schedules obtained by different models with respect to both a priori and a posteriori aircraft and passenger statistics.

3 MRP models

In order to highlight the interactions between metric, model, evaluation and data, we use different models to solve the Maintenance Routing Problem (MRP). For a formal definition of the problem, see Barnhart et al. (1998b) or Lan et al. (2006).

We use seven different models that we define in this section, namely:

RAMR'	maximize slack for minimal delay propagation using rerouting only;
RFSR'	minimize deviation from initial schedule for minimal delay propagation using retiming only;
RAMR'-RFSR'	iteratively solve RAMR' then RFSR'
IT_RR	maximize total slack using rerouting only;
MIT_RR	maximize minimum slack using rerouting only;
IT_RT	maximize total slack using retiming only;
IT_RT	maximize minimum slack using retiming only.

IT_RR, MIT_RR, IT_RT and IT_RT are based on the same underlying Uncertainty Feature Optimization model of Eggenberg et al. (2009). The approach is non-historical driven, in the sense that it solve a myopic deterministic problem that does not use any historical data. We describe the models in section 3.3.

`RAMR'`, `RFSR'` and `RAMR'-RFSR'` use historical data to estimate expected delays for each flight. These are used to evaluate delay propagation of a *string*, which is a feasible route for an aircraft (Barnhart et al., 1998b). These models minimize the propagated delay, which is thus our initial robustness metric.

The historical data is used to determine, for each string, the following values:

PDT_i	planned departure time of flight i
PAT_i	planned arrival time of flight i
ADT_i	actual departure time of flight i
AAT_i	actual arrival time of flight i
MTT	minimum turn time required to turn an aircraft
PTT_{ij}	planned turn time between flight leg i and j in the string
TDD_i	total departure delay of flight i
TAD_i	total arrival delay of flight i
$slack_{ij}$	the slack between flights i and j in the string
pd_{ij}	propagated delay from flight leg i to flight leg j in the string
IDD_i	independent departure delay of flight i
IAD_i	independent arrival delay of flight i

These constants satisfy the following set of relationships:

$$\begin{aligned}
PTT_{ij} &= PDT_j - MTT \\
TDD_i &= \max\{ADT - PDT, 0\} \\
TAD_i &= \max\{AAT - PAT, 0\} \\
slack_{ij} &= PTT_{ij} - MTT \\
pd_{ij} &= max\{TAD_i - slack_{ij}, 0\} \\
IDD_j &= TDD_j - pd_{ij} \\
IAD_j &= TAD_j - pd_{ij}
\end{aligned}$$

Given the independent arrival delay (IAD) of each flight, we compute the propagated delay between each pair of successive flights for any string, assuming the first flight of the string has zero departure delay (see Lan et al., 2006 for details). As d^s is the vector of IAD of all flight legs in string s, we denote the total propagated delay of string s by $f_s(d^s)$; it is computed using vector d^s as in Lan et al. (2006). Given a sample of N days, we compute the vector d^s_n of IAD for string s for each day $n \in N$, and use the following functions to determine the propagated delay pd_s of string s:

$$\begin{aligned}
&\texttt{H1}: \quad pd_s = \tfrac{1}{|N|} \textstyle\sum_{n \in N} f_s(d^s_n) \\
&\texttt{H2}: \quad pd_s = f_s(\tfrac{1}{|N|} \textstyle\sum_{n \in N} d^s_n)
\end{aligned}$$

`H1` corresponds to the arithmetic mean of the observed propagated delays over the N days; `H2` corresponds to the propagation of the average delays. In other words, for `H1`, we determine the propagated delay of string s on each day n (using the procedure of Lan et al., 2006), whereas for `H2`, we compute it only once, using, for each flight of a string, its average delay over the N days.

Finally, we list here the notation used throughout this section:

S	set of feasible strings, indexed by s;
F	set of flight legs, indexed by i or j;
P	the set of planes, indexed by p;
F_0	set containing the first flight of each string;
A	set of aircraft connections between two flights (i, j);
I	set of passenger connections between two flights denoted (i, j);
N	number of days in historical data;
M^+	set of initial states, indexed by m;
M^-	set of final states, indexed by m;
S_{m^+}	set of strings starting with initial state $m \in M^+$;
S_{m^-}	set of strings ending with final state $m \in M^-$;
pd_s	a proxy of total propagated delay of string s ;
tad_i^n	total arrival delay of flight i on day $n \in N$;
pd_{ij}^n	propagated delay from flight i to flight j on day $n \in N$ for $(i, j) \in A$;
d_i^n	independent arrival delay of flight i on day $n \in N$;
b_s^i	1 if string s covers flight $i \in F$, 0 otherwise;
b_s^m	1 if string s reaches the final state $m \in M^-$, 0 otherwise;
b_s^p	1 if string s is assigned to plane $p \in P$, 0 otherwise;
C	maximum absolute deviation between original and actual departure times of the entire schedule, in minutes;
c_s	absolute deviation (in minutes) between original and actual departure times for each flight in string s, i.e. flights such that $b_s^f = 1$;
δ_s	the total idle time in string s;
δ_s^{min}	the minimal idle time in string s.

3.1 Robust Airline Maintenance Routing

The Robust Airline Maintenance Routing (RAMR) model is an aircraft-centric model minimizing propagated delay by rerouting aircraft; see Lan et al. (2006). The model is based on strings, which are feasible routes satisfying the initial and final location requirements of the aircraft operating the string.

For each aircraft, we thus define the *initial state* and *final state* as the start and end points of a string, respectively. Both initial and final states are uniquely defined by an airport, a time and aircraft. Note that each aircraft has a unique initial state, but may have several candidate final states as plane swaps are allowed.

Note that Lan et al. (2006) define the sets M^+ and M^- as *maintenance stations* to model maintenance requirements. As our data does not contain maintenance information, we consider initial and final states as the unique maintenance stations, and suppose maintenance requirements are always satisfied.

In the string-based model, the binary decision variables are x_s, taking value 1 if string s is chosen in the optimal solution and 0 otherwise. Using this notation, the modified version of the Robust Airline Maintenance Routing (RAMR) model of Lan et al. (2006) is the following mixed-integer program:

$$z^*_{RAMR} = \max \sum_{s \in S} (x_s \times pd_s) \tag{1}$$

s.t. (2)

$$\sum_{s \in S} b^i_s x_s = 1 \qquad \forall i \in F \tag{3}$$

$$\sum_{s \in S^+_m} x_s = 1 \qquad \forall m \in M^+ \tag{4}$$

$$\sum_{s \in S^-_m} x_s = 1 \qquad \forall m \in M^- \tag{5}$$

$$x_s \in \{0, 1\} \qquad \forall s \in S \tag{6}$$

Objective (1) minimizes the total propagated delay using either `H1` or `H2` to derive pd_s. Constraints (3) ensure that each flight is covered by exactly one string, (4) ensures all the initial states are assigned to exactly one string and (5) ensures that each final state is covered by exactly one string.

Actually, formulation (1)-(6) typically contains a large set of optimal values. We thus derive model RAMR', which selects, among all optimal solutions of RAMR, the one with the largest total slack:

$$z^*_{RAMR'} = \max \sum_{s \in S} \left(x_s \times \sum_{i,j \in S} slack^s_{ij} \right) \tag{7}$$

s.t.

$$\sum_{s \in S} b^i_s x_s = 1 \qquad \forall i \in F \tag{8}$$

$$\sum_{s \in S^+_m} x_s = 1 \qquad \forall m \in M^+ \tag{9}$$

$$\sum_{s \in S^-_m} x_s = 1 \qquad \forall m \in M^- \tag{10}$$

$$\sum_{s \in S} (x_s \times pd_s) \leq z^*_{RAMR} \tag{11}$$

$$x_s \in \{0, 1\} \qquad \forall s \in S \tag{12}$$

Constraints (8)-(10) and (3)-(5) are identical, and the additional constraint (11) ensures that the solution of RAMR' is an optimal solution of RAMR. Solving RAMR' thus requires to solve RAMR first to get the value z^*_{RAMR}.

3.2 Robust Flight Schedule Retiming

Due to the lack of passenger data at hand, we cannot efficiently apply the connection based flight schedule retiming model of Lan et al. (2006), which minimizes the expected number of disrupted passengers. Instead, we formulate the Robust Flight Schedule Retiming (RFSR) model that minimizes the average total propagated delay. This model is equivalent to the one of AhmadBeygi et al. (2008); we do, however, not construct propagation trees as done in the original model.

In RFSR, the variables x_i correspond to the deviation of the departure time of flight i with respect to its original departure time; l_i and u_i are the lower and upper bounds of x_i,

respectively. These bounds limit the maximum retiming for a single flight, and we have $l_i \leq 0 \leq u_i$, a negative value of x_i meaning that the flight takes off earlier than originally planned. Decision variables y_{ij} ensure the new slack between flights i and j is consistent with the values of variables x_i and x_j. Finally, note that unlike the string-based model where the propagated delay is cumulative along a string, the delay propagation is considered for each connection independently; we denote pd_{ij}^n the delay propagation observed on day n for the flight connection $(i,j) \in A$.

RFSR is then given by the following linear program:

$$z^*_{RFSR} = \min \sum_{(i,j)\in A} \frac{1}{|N|} \sum_{n\in N} pd_{ij}^n \tag{13}$$

s.t.

$$tad_i^n \geq d_i^n \qquad \forall i \in F_0, \forall n \in N \tag{14}$$

$$tad_j^n \geq pd_{ij}^n + d_j^n \qquad \forall (i,j) \in A, \forall n \in N \tag{15}$$

$$tad_i^n \geq 0 \qquad \forall i \in F, \forall n \in N \tag{16}$$

$$y_{ij} = slack_{ij} - x_i + x_j \qquad \forall (i,j) \in A \cup I \tag{17}$$

$$y_{ij} \geq 0 \qquad \forall (i,j) \in A \cup I \tag{18}$$

$$pd_{ij}^n \geq tad_i^n - y_{ij} \qquad \forall (i,j) \in A, \forall n \in N \tag{19}$$

$$pd_{ij}^n \geq 0 \qquad \forall (i,j) \in A, \forall n \in N \tag{20}$$

$$l_i \leq x_i \leq u_i \qquad \forall i \in F \tag{21}$$

The objective (13) is to minimize the total average propagated delay. Constraints (14)-(16) are used to determine the total arrival delay of flight i for each day n (the first flight of each string has zero propagated delay by assumption).

Constraints (17) evaluate the new slack y_{ij} of each aircraft connection $(i,j) \in A$ and each passenger connection $(i,j) \in I$, excluding minimum turnaround and minimum connection times respectively. Constraints (18) ensure the non-negativity of all slacks: for the passenger connections, this implies no existing passenger connection is lost due to retiming, whereas for aircraft connections, this enforces the feasibility of the plane routings with respect to minimum turnaround times.

Finally, constraints (19) and (20) determine the delay propagating from flight i to flight j on day n, which has to be minimized. Note that delay propagation is only considered for aircraft connections, i.e. the delay propagation along strings; passenger connections between flights of different strings do not generate propagated delay.

RFSR directly determines the average delay propagation: we do not require to determine the values using `H1` or `H2`. As RFSR is minimizing the average propagated delay over the N days, it is similar to `H1` regarding the way historical information is used.

RFSR is equivalent to the model of AhmadBeygi et al. (2008), which is proved to find integer solutions of x. This formulation also contains a large set of optimal solutions; we thus derive model RFSR', which selects, among all optimal solutions of RFSR, the one that minimizes the changes with respect to the original schedule:

$$z^*_{RFSR'} = \min \sum_{i \in F} |x_i| \tag{22}$$

s.t.

$$\sum_{(i,j)\in A} \frac{1}{|N|} \sum_{n \in N} pd^n_{ij} \leq z^*_{RFSR} \tag{23}$$

$$tad^n_i \geq d^n_i \qquad \forall i \in F_0, \forall n \in N \tag{24}$$

$$tad^n_j \geq pd^n_{ij} + d^n_j \qquad \forall (i,j) \in A, \forall n \in N \tag{25}$$

$$tad^n_i \geq 0 \qquad \forall i \in F, \forall n \in N \tag{26}$$

$$y_{ij} = slack_{ij} - x_i + x_j \qquad \forall (i,j) \in A \cup I \tag{27}$$

$$y_{ij} \geq 0 \qquad \forall (i,j) \in A \cup I \tag{28}$$

$$pd^n_{ij} \geq tad^n_i - slack'_{ij} \qquad \forall (i,j) \in A, \forall n \in N \tag{29}$$

$$pd^n_{ij} \geq 0 \qquad \forall (i,j) \in A, \forall n \in N \tag{30}$$

$$l_i \leq x_i \leq u_i \qquad \forall i \in F \tag{31}$$

Objective (22) ensures the total deviation from the original schedule is minimized while constraint (24) ensures the optimality of the solution according to RFSR, which has to be solved first to determine z^*_{RFSR}. Constraints (24)-(31) are the same than (14)-(21).

Finally, we derive a model minimizing the propagation of average delays which corresponds to using historical data as done in `H2`: we use the average delay (estimated over the N days) to derive pd_{ij}, the propagated average delay from flight i to flight j. The formulation is as follows:

$$z^*_{RFSRH2} = \min \sum_{(i,j)\in A} pd_{ij} \tag{32}$$

s.t.

$$tad_i \geq \frac{1}{|N|} \sum_{n \in N} d^n_i \qquad \forall i \in F_0, \tag{33}$$

$$tad_j \geq pd_{ij} + \frac{1}{|N|} \sum_{n \in N} d^n_j \qquad \forall (i,j) \in A \tag{34}$$

$$tad_i \geq 0 \qquad \forall i \in F \tag{35}$$

$$y_{ij} = slack_{ij} - x_i + x_j \qquad \forall (i,j) \in A \cup I \tag{36}$$

$$y_{ij} \geq 0 \qquad \forall (i,j) \in A \cup I \tag{37}$$

$$pd_{ij} \geq tad_i - y_{ij} \qquad \forall (i,j) \in A \tag{38}$$

$$pd_{ij} \geq 0 \qquad \forall (i,j) \in A \tag{39}$$

$$l_i \leq x_i \leq u_i \qquad \forall i \in F \tag{40}$$

3.3 IT and MIT models

The non-historical driven models we use are derived from the models presented in Eggenberg and Salani (2009). These models consider the uncertainty implicitly by the means of Uncertainty Features (UF)s, which are structural properties of a solution that are shown to be improving the solution's robustness. In their application of Uncertainty Feature Optimization (UFO) to the plane routing problem, show that increasing the total slack (model `IT` for *idle time*) or the sum of minimal slack of each route (model `MIT` for *minimal idle time*) are improving the schedules' robustness as well as reducing the recovery costs in the case the solution is infeasible. The idle time as defined by Eggenberg and Salani (2009) is the slack time between two successive flights

without the minimum turnaround time, i.e. it is equivalent to the slack as defined for the RAMR and RFSR models.

The model maximizing the number of plane crossings (model `CROSS`) in Eggenberg and Salani (2009) is showing reducing the recovery costs, but the proposed formulation is actually decreasing slack, making the solutions less robust. We therefore consider only adapted versions of the models `IT` and `MIT`.

The original models of Eggenberg and Salani (2009) aim at maximizing the UF while keeping the total (absolute) deviation between original planed departure times and the new departure times bounded by a constant C (in minutes); in the case no retiming is allowed, we obviously set $C = 0$.

A final state $m \in M^-$ models the fleet positioning requirements at the end of the scheduling window, and is uniquely defined by a location (airport), a latest *ready* time (in opposition to landing time) and a plane type.

The decision variables of the problem are $x_s \in \{0, 1\}$, $s \in S$, being 1 if string s is selected in the solution and 0 otherwise. The uncertainty features `IT` and `MIT` are the following linear functions:

$$\mu_{\mathtt{IT}}(\mathbf{x}) = \sum_{s\in S} \delta_s x_s$$

$$\mu_{\mathtt{MIT}}(\mathbf{x}) = \sum_{s\in S} \delta_s^{\mathtt{min}} x_s$$

The UFO formulation of the plane routing problems is then following mixed-integer program, where $\mu(\mathbf{x})$ is either $\mu_{\mathtt{IT}}(\mathbf{x})$ or $\mu_{\mathtt{MIT}}(\mathbf{x})$:

$$z_{\mathtt{UFO}} = \max \mu(\mathbf{x}) \tag{41}$$

s.t.

$$\sum_{s\in S} b_s^i x_s = 1 \qquad \forall i \in F \tag{42}$$

$$\sum_{s\in S} b_s^m x_s = 1 \qquad \forall m \in M^- \tag{43}$$

$$\sum_{s\in S} b_s^p x_s \leq 1 \qquad \forall p \in P \tag{44}$$

$$\sum_{s\in S} c_s x_s \leq C \tag{45}$$

$$x_s \in \{0, 1\} \qquad \forall s \in S \tag{46}$$

The formulation (41)-(46) slightly differs from the formulation in Eggenberg and Salani (2009), which allows for flight cancellation and also consider airport capacities; equivalence between the models is achieved when all flight cancellation costs and all airport capacities are infinite. Constraints (42) and (43) ensure that each flight and each final state are covered are covered by exactly one route, respectively. Constraints (44) ensure that each plane is affected to at most one route, and constraint (45) limits the maximum deviation between original and new schedule.

To solve problem (41)-(46) we the column generation algorithm described in Eggenberg and Salani (2009), using the *recovery networks* described in Eggenberg et al. (forthcoming). A recovery network is a graph containing all feasible routes for a particular aircraft, and the pricing problem of the column generation algorithm corresponds to a Resource Constrained Elementary Shortest Path Problem (ESPPRC) on the recovery networks, which are generated using a dynamical algorithm. Note that a recovery network for a specific plane contains only flights the plane is allowed to cover. When restricted to the flights originally affected the same plane the recovery network is associated with, we forbid plane swaps; this is what we use to solve the plane retiming only problem, i.e. for IT_RT and MIT_RT. To solve the rerouting only

problem, i.e. IT_RR and MIT_RR, the set of coverable flights for a plane is the set of all flights originally assigned to planes of the same fleet, and the total deviation constant is set to $C = 0$.

4 Case study from a real airline

We compare the different models defined in section 3 using data from an airline operating in the US, central America and towards Europe as well. We are provided with 2 months of data. The first month, namely February 2008, is used for delay estimation and the second, March 2008, for validation.

Unfortunately, the passenger data cover only a few weeks of operations; our choice of not considering passenger-centric models is closely related to the data at hand, as it does not allow to derive statistically relevant information. Furthermore, we believe that using the same data for estimation of the probabilities and evaluation of a solution leads to an unfair comparison. We thus use only non-passenger-centric models, but use the passenger statistics to evaluate the models.

We use the data of the month of March 2008 for computation and evaluation of the different models. As the schedule is not cyclic, we solve each day of operation independently using the different models. We then compare the original schedule and the new schedules generated by the different models using the real observed independent arrival delays (IAD).

Our strongest assumption with respect to the airline's real operations is that the minimum turnaround time is uniquely fleet dependent, as in most of the literature. This is, however, not the case in the airline's data: minimal turnaround time depends on the airport and on the flight preceding and the one following the grounding, respectively. Looking more in details at the data, however, we see that the airline is almost systematically underestimating block time of the flights, but the real observed ground time is, when critical, systematically lower than the planned minimum turnaround time, namely around 30 minutes in average, independently of fleet, location and previous and following flights.

Interestingly, we observe that the increased turnaround time for two specific flights involve precisely those flights that are most delayed. Figure 1 shows the distribution of the difference between scheduled and real block-time. We observe that the majority of the observed values are negative, meaning that the planned block time is lower than the observed block-time. Proportionally, about 60% of the flights have underestimated block time, the average underestimation over almost 7,000 flights is 6.48 minutes, i.e. the total delay incurred by block time underestimation only is almost 45,000 minutes. For the minimum turnaround times, we consider connections for which we observe propagated delay, i.e. when the connection is tight. Figure (1) shows that the difference between observed and planned minimum turnaround times is positive, i.e. minimum turnaround is over-estimated, in almost 70% of the tight connections. The total amount of over-estimated minutes is more than 4100 minutes for less than 1050 flights.

We conclude from these observations that the increased turnaround time is the airline's response to observed delays, i.e. their strategy to make their schedule more robust.

We thus seek a reasonable value of the *real* turnaround time: Figure 2 shows the observed turnaround time for tight connections, i.e. where propagated delay is observed, without differentiating fleet; looking at the average value, we conclude that a 30 minutes minimum turnaround time is a reasonable value.

We thus use the 30 minutes as minimum turnaround time for both our models and the simulations: when evaluating a schedule with observed delays, all planes require 30 minutes turnaround time between two flights for all schedules. When delay occur, the departure of a flight is thus the maximum between the scheduled departure and the real arrival time plus 30 minutes.

Finally, in our experiments, we assume that the minimum passenger connection time is 30 minutes. For passengers, we differentiate *lost* passengers from *disrupted* passengers: the former are passengers who have a connection of less than 30 minutes in the original schedule, without delay. Lost passengers might occur in retiming models when passenger connections are

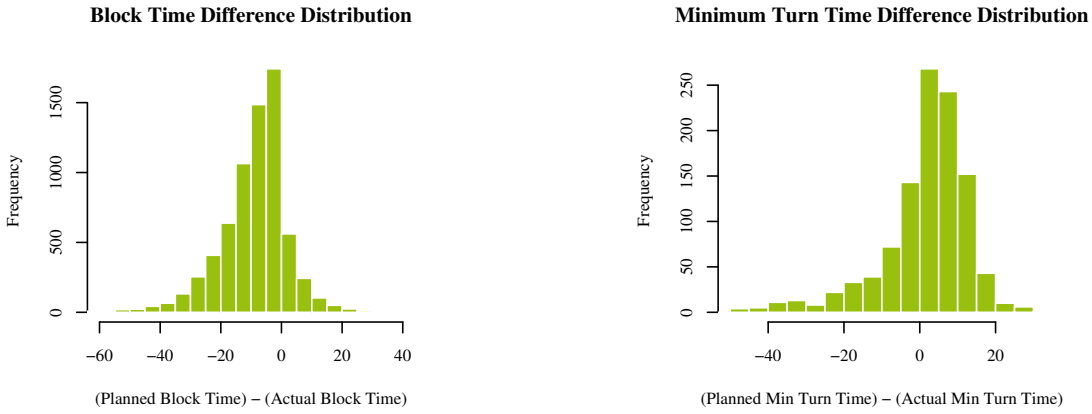

Figure 1: Distribution of the difference between planned and observed block-times (left) and minimum turnaround times (right).

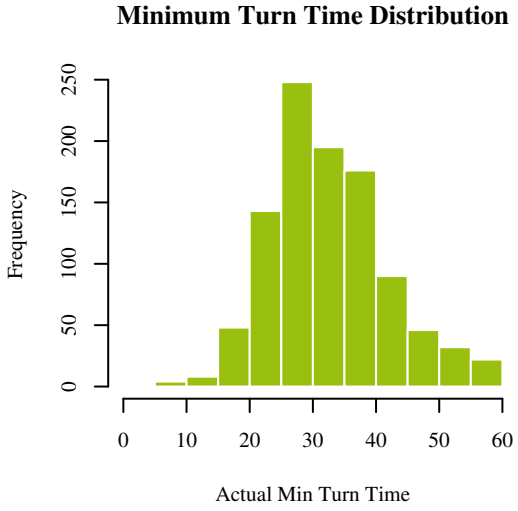

Figure 2: Distribution of the observed turnaround time for connections that experience delay propagation; the average turnaround time is 30 minutes.

not explicitly considered. In that case, we assume that the passengers are not able to buy a ticket in the first place, and are not considered when computing delay statistics. The latter are passengers who have an original connection time larger than 30 minutes but miss a connection due to delays. When computing passenger delay statistics, we try to re-route the disrupted passengers according to a first-come-first-served (FCFS) strategy based on the the algorithms in Bratu (2003) and Bratu and Barnhart (2005): we assume a maximum delay of 12 hours for passengers, overnight stay is not allowed, and only itineraries of at most two flights are rerouted; all passengers that could not be rerouted are considered canceled. Canceled passengers are not considered for delay estimation, unlike done by Bratu (2003) and Bratu and Barnhart (2005), who assign a constant delay to them.

Figure 3 shows the evolution of observed propagated delay and the corresponding computed total passenger delay for March, 2008. We see two main peaks of delays between the 7^{th} and

14^{th} day and a smaller one with respect to propagated delay on day 23. Propagated and total passenger delays are closely related, although not identical, which is non-trivial. There are two explanations for this. Firstly, the total passenger delay is multiplied by the number of passengers: there is thus a multiplicative effect between propagated and passenger delays. Furthermore, due to slack, a flight might be delayed even though it generates no delay propagation: the passengers on this flight generate a non-zero delay although delay propagation is zero. This shows that although passenger and propagated delays are not independent, they are not perfectly correlated either.

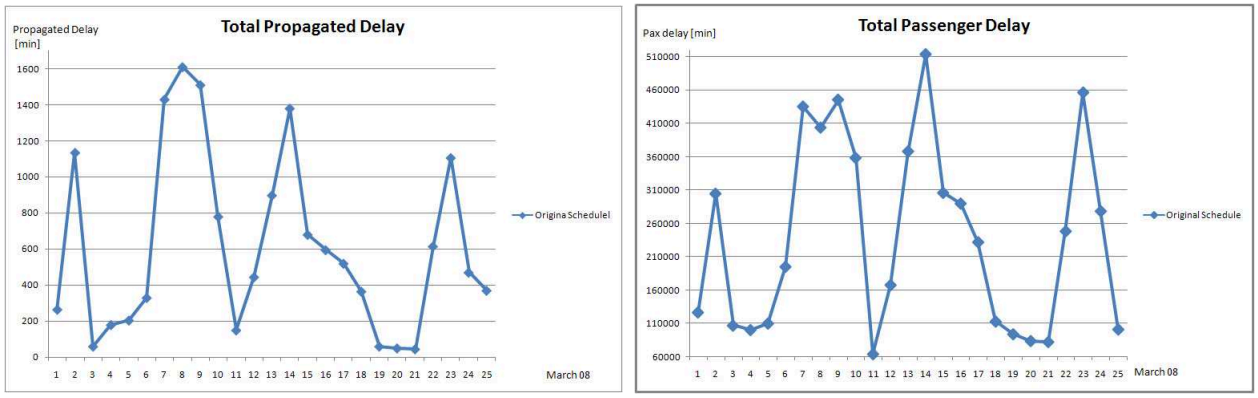

Figure 3: Observed propagated delay (on the left) and total passenger delay (right) for the original schedule over 25 days of operations in March, 2008.

For the results, we use the data from March, 1^{st}, 2009 to March, 25^{th}, 2009. We compare the solutions obtained by the following models:

`Original`	the original airline's schedule, using over-estimated turnaround times to avoid delay propagation
`RAMRB'_H1`	model minimizing propagated delay in a first stage, and maximizing slack in a second stage, with re-routing only
`RAMRB'_H2`	model minimizing propagated delay in a first stage, and maximizing slack in a second stage, with re-timing only
`RFSR'_H1`	model minimizing propagated delay in a first stage, and minimizing the total deviation from the original schedule in a second stage, with re-routing only
`RFSR'_H2`	model minimizing propagated delay in a first stage, and minimizing the total deviation from the original schedule in a second stage, with re-timing only
`RAMR'-RFSR'_H1`	model for which solve `RAMR'_H1` first, and then solve `RFSR'_H1` using the routing solution obtained by `RAMR'_H1`
`RAMR'-RFSR'_H2`	model for which solve `RAMR'_H2` first, and then solve `RFSR'_H2` using the routing solution obtained by `RAMR'_H2`
`IT_RR`	model maximizing total slack allowing for re-routing only
`IT_RT`	model maximizing total slack allowing for re-timing only
`MIT_RR`	model maximizing the sum of minimum slacks for each route, allowing for re-routing only
`MIT_RT`	model maximizing the sum of minimum slacks for each route, allowing for re-timing only

Note that for models using `RFSR'`, we use $-15 \leq x_i \leq 15$, i.e. retiming of a single flight is limited to a time window starting 15 minutes before and ending 15 minutes after its original departure time.

We hereafter compare the results according to the four different points robustness depends on.

4.1 A priori and a posteriori results

A priori metrics. First of all, we compare the different models according to the following priori statistics:

- total slack
- total retiming
- average lost connections
- average lost passengers
- maximum lost connections
- maximum lost passengers

The total slack is the a priori robustness metric, the other metrics allow to quantify the *price of robustness* (Bertsimas and Sim, 2004), i.e. the loss of revenue to gain robustness.

The aggregated a priori metric over the 25 days of operations for the different models are reported in Table 1. All statistics are a daily average, and lost connections/passengers are such that the connection time is lower than 30 minutes.

A posteriori metrics. To compare the performance of the different models, we evaluate all the schedules on 25 real days of operation. We compare the average value on the 25 days of the following metrics:

Model	Original	IT_RR	IT_RT	MIT_RR	MIT_RT	RAMR'_H1	RAMR'_H2
Tot. Slack [min]	8871.96	9494.96	9731.44	9154.36	9759.16	9699.96	9753.16
Tot. Retiming [min]	0.00	0.00	898.96	0.00	1493.68	0.00	0.00
Avg. Lost Connections	3.04	3.04	3.08	3.04	5.12	3.04	3.04
Avg. Lost pax	5.84	5.84	6.12	5.84	14.68	5.84	5.84
Max. Lost Connections	7	7	7	7	9	7	7
Max. Lost pax	18	18	18	18	43	18	18

Model	RFSR'_H1	RFSR'_H2	RAMR'-RFSR'_H1	RAMR'-RFSR'_H2
Tot. Slack [min]	9577.44	8921.68	10319.16	9800.12
Tot. Retiming [min]	1634.68	120.24	1469.16	107.88
Avg. Lost Connections	3.00	3.00	3.04	3.04
Avg. Lost pax	5.60	5.80	5.80	5.84
Max. Lost Connections	8	7	7	7
Max. Lost pax	20	18	18	18

Table 1: Average a priori statistics over 25 days of operations in March, 2008 for the different models.

- propagated delay
- total arrival delay
- number of disrupted passengers
- number of canceled passengers
- total passenger delay (including delays after rerouting the disrupted passengers)
- non-disrupted passenger delay
- disrupted passenger delay

Table 2 displays the average statistics for these metrics. We recall that disrupted passengers are those whose original connection time was larger than 30 minutes, but the real connection time is lower than 30 minutes due to delays; canceled passengers are disrupted passengers that could not be rerouted within 10 hours of delay.

Figures 4 and 5 show the daily values for the total propagated delay and the number of disrupted passengers for the most relevant models.

4.2 Sensitivity to metrics

When comparing metrics, we compare the different models with respect to their objective and the initial robustness metric we focus on, namely the propagated delay. Models RAMR' and RFSR' minimize propagated delay as a primary objective. IT and RAMR' maximize slack (as a secondary objective for RAMR'), models MIT maximize the minimal slack and models RFSR' minimize the deviation from the original schedule as a secondary objective. The performance metrics we to consider here are the metrics we optimize on, i.e. total slack and propagated delay.

We see from Table 1 that the different objectives lead to different values for the total slack, meaning that the slack is distributed differently depending on the initial objective. Note that all models increase the slack with respect to the original schedule.

Looking at Table 2, we see that the models with lowest propagated delay are indeed the ones minimizing propagated delay, mainly RFSR'_H1 and RAMR'-RFSR'_H1.

This shows that the best results are indeed obtained with the specific metric, i.e. the models minimizing propagated delay. However, as we discuss in the following sections, the optimized metric is not the only relevant factor.

Model	Original	IT_RR	IT_RT	MIT_RR	MIT_RT	RAMR'_H1	RAMR'_H2
Propagated Delay [min]	610.16	575.08	535.44	593.96	439.12	538.16	579.76
15 min on-time performance [%]	76.44	76.57	76.90	76.54	77.85	76.82	76.56
60 min on-time performance [%]	97.07	97.17	97.24	97.09	97.32	97.17	97.07
Arrival Delay [min]	3,108.64	3,074.16	3,042.32	3,093.24	2,955.20	3,039.24	3,080.44
Disrupted pax	22.44	22.56	18.60	22.48	26.32	25.08	23.96
Canceled pax	25.00	26.24	22.20	25.20	24.00	26.28	25.48
Total pax delay	239,377	237,439	233,724	238,635	230,102	236,862	239,568
Avg. non-disrupted pax delay	15.02	14.90	14.74	14.98	14.43	14.87	15.04
Avg. disrupted pax delay	188.89	189.23	159.74	188.64	171.25	170.05	175.06

Model	RFSR'_H1	RFSR'_H2	RAMR'-RFSR'_H1	RAMR'-RFSR'_H2
Propagated Delay [min]	400.24	550.36	364.64	536.20
15 min on-time performance [%]	78.28	76.94	78.46	76.92
60 min on-time performance [%]	97.42	97.10	97.34	97.05
Arrival Delay [min]	2,917.52	3,053.24	2,884.92	3,041.16
Disrupted pax	35.80	25.96	41.92	25.20
Canceled pax	24.52	24.32	29.36	24.88
Total pax delay	230,037	236,537	229,842	237,474
Avg. non-disrupted pax delay	14.31	14.83	14.26	14.90
Avg. disrupted pax delay	174.52	172.90	167.17	170.26

Table 2: Average delay statistics over 25 days of operations in March, 2009 for the different models; canceled passengers are disrupted passengers that could not be re-routed within a maximum of 10 hours delay and no overnight using a first-come-first-served (FCFS) algorithm.

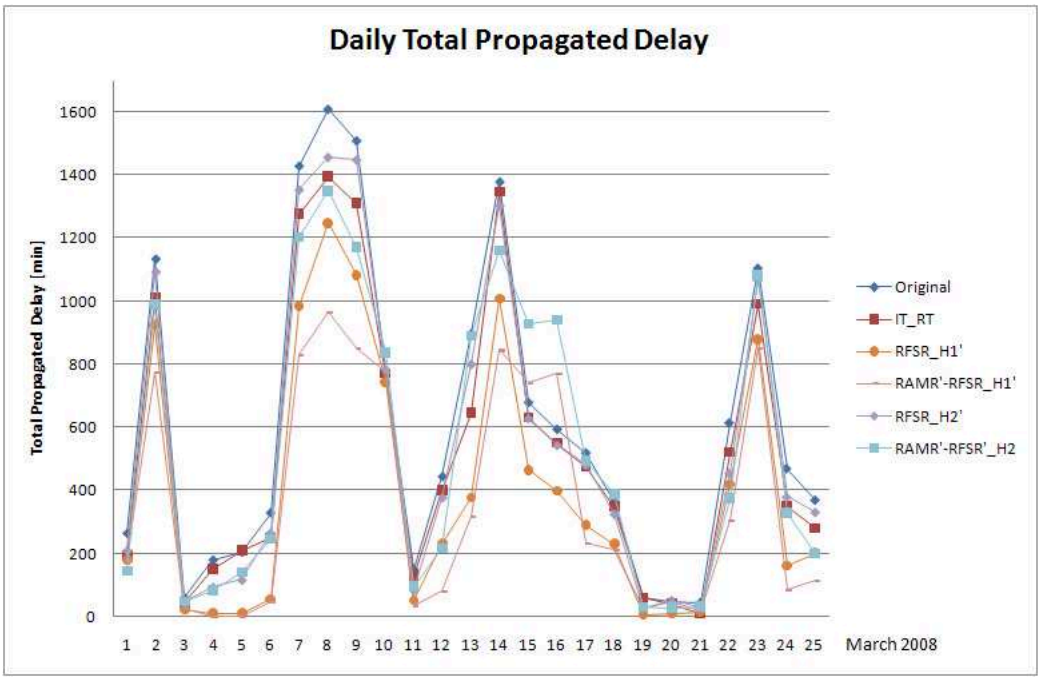

Figure 4: Daily total propagated delay for different models for 25 days of operations in March, 2008.

4.3 Sensitivity to models

The different models are divided in three classes: rerouting-only (IT_RR, MIT_RR, RAMR'_H1 and RAMR'_H2), retiming-only (IT_RT, MIT_RT, RFSR'_H1 and RFSR'_H2) and both rerouting and

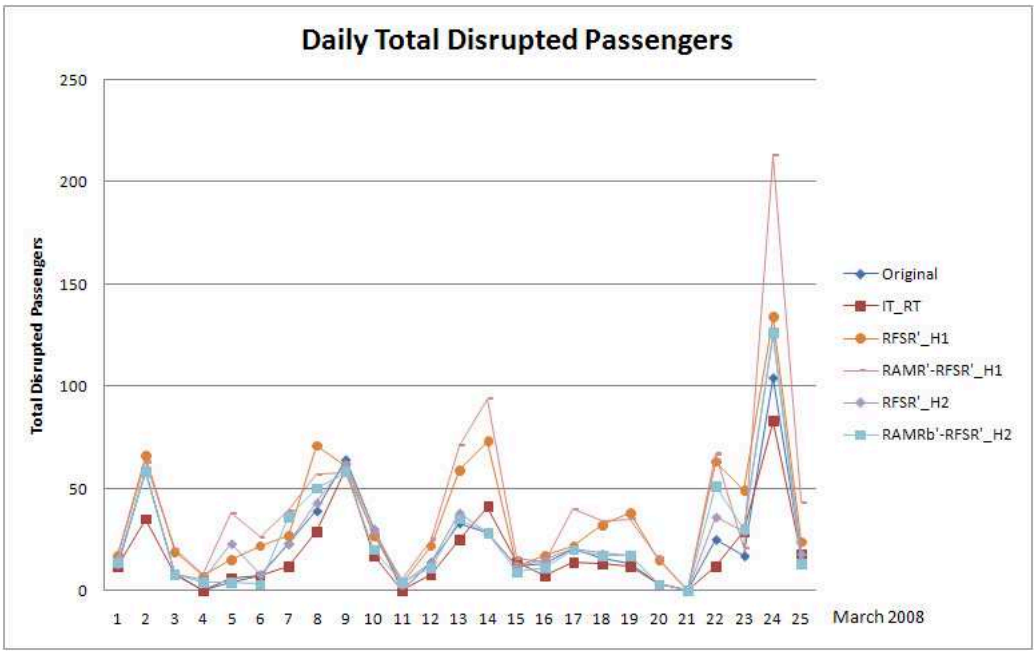

Figure 5: Daily number of disrupted passengers delay for different models on the 25 days of operations in March, 2008.

retiming (`RAMR'-RFSR'_H1` and `RAMR'-RFSR'_H2`).

Table 1 shows that the rerouting-only models have, as expected, 0 retiming and thus the lost connections and passengers are equal to `Original`. Remarkably, `Original` has non-zero lost connections; this is because some connections have less than 30 minutes connection time even in the original schedule.

We observe that, a priori, the retiming models `IT_RT` and `IT_RT` are increasing the number of lost connections and thus lost passengers; the reason is that these models do not consider passenger connections explicitly as does `RFSR'`. Thanks to the explicit consideration of all connections, even those with less than 30 minutes connection in the original schedule, models `RFSR'` allow to reduce the number of lost passengers.

Clearly, for rerouting-only, `RAMR'_H1` leads to solutions with higher slack than `IT_RR` and `MIT_RR`. This illustrates that, although both models maximize slack, the way it is modeled induces significant differences in the final solution. For the retiming-only models, `IT_RT` and `MIT_RT` find solutions with higher slack than `RFSR'_H1` and `RFSR'_H2`. The main reasons are that (a) model `RFSR'` does not explicitly maximize slack, but minimizes the total deviation from the original schedule and (b) `RFSR'` explicitly considers the passenger connections, which reduces the solution space compared to the solution space of `IT_RT` and `MIT_RT`. The highest slacks are obtained with models `RAMR'-RFSR'_H1` and `RAMR'-RFSR'_H2`, for which both rerouting and retiming are considered, i.e. the feasible solution space is the largest.

The retiming models maximizing slack lead to solutions with higher slack than rerouting-only models. This is due to the fact that the retiming allows to extend the operation period up to 30 minutes in our case (the first flight departs 15 minutes earlier, the last 15 minutes later) and hence retiming allows an additional 30 minutes slack per string. This potential for additional slack is, however, not exploited by models `RFSR'_H1` and `RFSR'_H2`, as the objective is not maximal slack.

In terms of performance with respect to delay propagation, we observe the rerouting-only models lead to solutions with higher propagated delay than the retiming models. The reason is that, thanks to the retiming, the slack can be more specifically allocated where it is required than for rerouting-only models. However, we see that some rerouting-only models achieve lower

delay propagation than some retiming models. This is mainly the case for models using historical data according to H2; we discuss this issue in section 4.5

4.4 Evaluation on different performance metrics

In this section, we compare the performance of the solutions according to other robustness metrics than propagated delay to show the correlation between the original robustness metric and the others. We use two types of performance metrics, namely aircraft-based metrics and passenger-based metrics.

As discussed in the previous section, the original robustness metric we use is the propagated delay, for which the best results are obtained by the specific models minimizing the expected propagated delay. Furthermore, looking at the other aircraft statistics (15 and 60 min on-time and arrival delay), we observe a positive correlation in performance: a good solution in terms of delay propagation is also good according to the other aircraft-based metrics. Indeed, solutions with low propagated delay also have high on-time performance and low arrival delay. Model RAMR'-RFSR'_H1 is the best according to the aircraft-based metrics. However, the correlation is not perfect: for example, model Original is the worst for almost all metrics, especially for propagated delay, but RAMR'-RFSR'_H2 has actually a lower 60-minute on-time performance, although it has 15.7% less propagated delay.

A traditional performance metric for airlines is the the 15-minute on-time performance. We observe that retiming models increase it more substantially than rerouting-only models, especially with models RFSR'_H1 and RAMR'-RFSR'_H1. Indeed, with model RAMR'-RFSR'_H1, the 15 min on-time performance is increased by 2%: as discussed in Lan et al. (2006), an increase of 1.6% of the Department of Transportation (DOT) on-time arrival rate (i.e. the 15 min on-time performance) is sufficient for any top 5 airline to gain at least one rank in the DOT ranking. For the 60-minutes on-time performance, however, differences are smaller, all models leading to solutions between 97.05% and 97.42%.

The conclusion, when restricting to the aircraft metrics, is that retiming is more efficient than rerouting only. Furthermore, we see that using historical data to minimize propagated delay with model H1 always achieves the better results than the equivalent non-historical driven models: the reduction of propagated delay for the rerouting-only is 6.4% and 8.9% for the retiming models.

We now extend the performance comparison on passenger statistics. We exclude model MIT_RT form the comparison, as it has a significantly higher number of lost passengers compared to the other models. For the other models however, the total number of passengers is similar: the largest difference in number of lost passengers on a daily average is 0.62.

Remarkably, the best model according to the aircraft statistics, i.e. RAMR'-RFSR'_H1, is the one with highest number of disrupted and canceled passengers: compared to IT_RT, there are 125.4% more disrupted passenger and 32.3% more canceled passsengers. In absolute numbers, there are 7.16 more canceled passengers on a daily average; assuming that the 0.62 passengers that lost in IT_RT are canceled in RAMR'-RFSR'_H1, the additional number of canceled passengers is thus 6.54, which still corresponds to a 29.5% increase.

The total passenger delay is smallest for RAMR'-RFSR'_H1, which is due to the fact that canceled passengers do not account any delay. This illustrates the concept of negatively correlated performance metrics: by increasing the number of canceled passengers, we are able to reduce the total passenger delay. Similarly, the best solution according to delay propagation, on-time performance and total arrival delay is also the best according to total passenger delay, but the worst according to number of disrupted and number of canceled passengers.

This clearly shows that the way a solution's performance is evaluated dramatically changes its ranking, and that the concept of *best* solution is relative. Additionally, there is no intuitive way to balance the different metrics into a unique weighted performance metric. This implies that the absolute performance is either an arbitrary choice of certain metrics or that no absolute best solution exists.

4.5 Sensitivity to data

In this section, we focus on the differences in the way historical data is used for a same model. We thus mainly compare solutions using the average propagated delay H1 against models using the propagation of average delays H2.

As we see from Tables 1 and 2, there are significant differences for models RAMR', RFSR' and RAMR'-RFSR' depending on the used historical model.

Indeed, for the different models, using historical data according to H1 leads solutions with lower propagated delay than those obtained using H2. When historical data is a good representation of the real delay, this is reflects a well known principle in stochastic optimization called the *Value of Stochastic Solution* (VSS): the solution minimizing the average cost on the sum of a set of scenarios, as used in H1, has lower value than the sum of the average values, as in H2 (Birge and Louveaux, 1997).

We see that using historical data in an appropriate way allows to reduce delay propagation and the correlated metrics (on-time performance, arrival delay) compared to non-historical driven methods. Clearly, this is not the case for H2: RAMR'_H2 has higher propagated delay than the non-historical equivalent, IT_RR. This also holds for the retiming models, comparing RFSR'_H2 and IT_RT.

The easiest and most intuitive approach is to compute average delays for all flights and minimizing delay propagation accordingly; this is exactly what is done in H2. Alas, models using the data as H1 are computationally much harder in general. Indeed, model (13)-(21), using data as in H1, requires $(|\ N\ | - 1) \times (|\ F_0\ | + 3\ |\ A\ | + |\ F\ |)$ additional constraints than model (22)-(31), which uses model H2; the increase in number of variables is also of the order of $|\ N\ |$. The complexity thus depends on the sample size, which is crucial, as a too small sample of historical data does not guarantee a statistically relevant representation of the real delays.

The differences in the way historical data is used explains why model RAMR-RFSR_H1 has such a higher number of disrupted and canceled passengers. Indeed, it is the model that captures best the delay propagation and protects accordingly. In particular, as the delay propagation is performed using model H1, a unique occurrence of a huge delay on one flight accounts as much as many occurrences of small delays (or more). The model thus protects against such cases adding slack at more specific places than non-historical or H2 models. The consequence is that more flights are retimed, as show the total retiming statistics in Table 1. Furthermore, the retiming is limited due to passenger connections that have to be ensure, which implies that most of them are tight, i.e. $y_{ij} = 0$ for more passenger connections $(i, j) \in I$. Therefore, when delays occur, passenger connections are more likely to be lost in model RAMR-RFSR_H1, explaining the increased number of disrupted passengers.

The interesting question is where the differences between the models occur. We thus look at the daily values of the total propagated delay and the number of disrupted passengers in Figures 4 and 5 respectively.

On Figure 4, all rerouting models using historical data (even RAMR'_H1 and RAMR'_H2 which are not displayed in Figure 4) are performing worst with respect to propagated delay on March 15^{th} and 16^{th},. On these two days, some flights have much higher delay than expected from historical data: these flights are considered as reliable and used for tighter connections which did not exist in the original routing. On days 15 and 16, however, these reliable flights are delayed, which explains the high delay propagation with respect to the non-rerouting models. This is a typical example of the impact of an erroneous delay estimation on the routing decision.

Consider now Figure 5, showing the number of disrupted passengers for each day. The curves no longer exhibit peaks as clearly as in Figure 4. Indeed, the best models in terms of delay propagation are models RAMR'_H1 and RAMR'-RFSR'_H1, which are clearly the models with highest passenger cancellation. This holds both for days with high delays, as for example in days 12-14, than days with low delays, as in days 17-20. Model IT_RT is in general performing better in terms of total disrupted passengers, although it has higher delay propagation. As discussed previously, this is due to the fact that a high focus on delay propagation implies tighter passenger connections, which thus increases the probability of missed connections.

5 Conclusion

In this paper, we focus on the definition of robustness and show that it depends on four key factors: metrics, models, evaluation and data. We use a case study on a real airline to illustrate the importance of the different factors. We focus on the robust maintenance routing problem aiming at minimal delay propagation as an a priori robustness metric.

We show that if we use different metrics such as maximizing slack, solutions are significantly different. We also show that different models with the same objective lead significant differences. In particular, we show that, in average, retiming models perform better than rerouting models, as it allows for more slack and thus more delay absorption. However, we show that no solution is globally the best when evaluating the solution on different performance metrics. Indeed, we show that aircraft-based performance metrics and passenger-based performance metrics are negatively correlated.

Finally, we show that models using historical data are sensitive to the way the data is exploited. However, such models have the best potential when the data is representative of the real uncertainty and the historical data is exploited in an appropriate way.

Interestingly, in our simulations, we observe that the most intuitive model to exploit historical data is not better than non-historical based methods. Indeed, the easiest choice is to evaluate delay propagation by first estimating average delays for each flight from historical data and then computing propagation of these average delays, which leads to slightly less robust solutions in our results. Unfortunately, models using historical data more adequately are more complex and therefore limit the size of the solvable problems.

The main conclusion of this paper is that it is crucial for airlines to understand the relations between their scheduling objectives and the performance metrics they want to improve: if the airline aims at improving a specific performance metric, then historical-driven approaches are certainly better, provided the historical data is a reliable estimator, the data is well exploited and the model remains solvable. In the other cases, non-historical approaches are certainly the better choice.

This work should be extended by performing a more extensive study on different airlines with different schedules. Indeed, the data we use for our simulations comes from a unique airline which has a specific structure and a low number of connecting passenger, which may not be representative for large US carriers. Furthermore, one should consider testing historical models using simultaneously aircraft, crew and passenger data to see whether it is possible to exploit additional information, both in terms of computational complexity and solution quality, or if using implicit approaches as UFO (Eggenberg et al., 2009) is a better compromise.

6 Acknowledgments

We want to thank Jeppessen for providing the data of the airline, and their useful comments on both the airline's as well as the data structures.

References

Ageeva, Y. (2000). *Approaches to incorporating robustness into airline scheduling*, Master's thesis, Massachusetts Institute of Technology.

AhmadBeygi, S., Cohn, A. and Lapp, M. (2008). Decreasing airline delay propagation by re-allocating scheduled slack, *Technical report*, University of Michigan.
URL: *http://www.agifors.org/award/submissions2008/Ahmadbeygi_paper.pdf*

Annual Report 2008 (2008). International Air Transport Association.
URL: *http://www.iata.org*

Barnhart, C., Boland, N., Clarke, L., Johnson, E., Nemhauser, G. and Shenoi, R. (1998b). Flight string models for aircraft fleeting and routing, *Tranportation Science* **32**(3): 208–220.

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

Birge, J. R. and Louveaux, F. (1997). *Introduction to Stochastic Programming*, Springer.

Bratu, S. (2003). *Airline Passenger On-Time Schedule Reliability: Analysis, Algorithms and Optimization Decision Models*, PhD thesis, Massachusetts Institute of Technology.

Bratu, S. and Barnhart, C. (2005). An analysis of passenger delays using flight operations and passenger booking data, *Air Traffic Control Quarterly* **13**(1).

Bratu, S. and Barnhart, C. (2006). Flight operations recovery: New approaches considering passenger recovery, *J. Sched.* **9**: 279–298.

Burke, E., DeCausmaecker, P., Maerea, G. D., Mulderc, J., Paelinckc, M. and Berghed, G. V. (forthcoming). A multi-objective approach for robust airline scheduling, *Computers & Operations Research* **doi:10.1016/j.cor.2009.03.026**.

Clausen, J., Larsen, A. and Larsen, J. (forthcoming). Disruption management in the airline industry - concepts, models and methods, *Computers & Operations Research* **doi:10.1016/j.cor.2009.03.027.**
URL: *http://www.agifors.org/award/submissions2008/Weide_paper.pdf*

Eggenberg, N. and Salani, M. (2009). Uncertainty feature optimization for the airline scheduling problem, *Technical Report TRANSP-OR 090105*, Ecole Polytechnique Fédérale de Lausanne, Switzerland.

Eggenberg, N., Salani, M. and Bierlaire, M. (2009). Uncertainty feature optimization: an implicit paradigm for problems with noisy data, *Technical Report TRANSP-OR 080829*, Ecole Polytechnique Fédérale de Lausanne, Switzerland.

Eggenberg, N., Salani, M. and Bierlaire, M. (forthcoming). Constraint-specific recovery networks for solving airline recovery problems, *Computers & Operations Research* **doi:10.1016/j.cor.2009.08.006**.

Ehrgott, M. and Ryan, D. M. (2000). Bicriteria robustness versus cost optimization in tour of duty planning at Air New Zealand, *25th Annual Conference of Operations Research Society of New Zealand*.
URL: *http://www.orsnz.org.nz/conf35/papers/MatthiasEhrgott.pdf*

FAA Aerospace Forecast Fiscal Years 2008-2025 (2008). Federal Aviation Administration.
URL: *http://www.faa.gov*

Kang, L. S. (2004). *Degradable Airline Scheduling: an Approach to improve Operational Robustness and differenciate Service Quality*, PhD thesis, Massachusetts Institute of Technology.

Kohl, N., Larsen, A., Larsen, J., Ross, A. and Tiourine, S. (2007). Airline disruption management - perspectives, experiences and outlook, *Journal of Air Transport Management* **13**(3): 149–162.

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

Listes, O. and Dekker, R. (2005). A scenario aggregation-based approach for determining a robust airline fleet composition for dynamic capacity allocation, *Transportation Science* **39**(3): 367–382.

Rosenberger, J., Johnson, E. and Nemhauser, G. (2004). A robust fleet assignment model with hub isolation and short cycles, *Transportation Science* **38**(3): 357–368.

Rosenberger, J., Schaefer, A., Golldsman, D., Johnson, E., Kleywegt, A. and Nemhauser, G. (2003a). A stochastic model of airline operations, *Transportation Science* **36**(4): 357–377.

Schaefer, A., Johnson, E., Kleywegt, A. and Nemhauser, G. (2005). Airline crew scheduling under uncertainty, *Transportation Science* **39**(3): 340–348.

Shebalov, S. and Klabjan, D. (2006). Robust airline scheduling: Move-up crews, *Tranportation Science* **40**(3): 300–312.

Smith, B. and Johnson, E. (2006). Robust airline fleet assignment: Imposing station purity using station decomposition, *Transportation Science* **40**(4): 497–516.

Weide, O. (2009). *Robust and Integrated Airline Scheduling*, PhD thesis, The University of Auckland, New Zealand.

Yen, J. W. and Birge, J. R. (2006). A stochastic programming approach to the airline crew scheduling problem, *Transportation Science* **40**: 3–14.

Your Flight Has Been Delayed Again (2008). Joint Economic Committee.
URL: *http://jec.senate.gov*