

Robust Airline Schedule Planning: Minimizing Propagated Delay in an Integrated Routing and Crewing Framework

Michelle Dunbar, Gary Froyland

School of Mathematics and Statistics, University of New South Wales, Sydney NSW 2052, Australia.

Cheng-Lung Wu

School of Aviation, University of New South Wales, Sydney NSW 2052, Australia.

Abstract

To retain a degree of tractability, the airline scheduling problem has traditionally been sequentially decomposed into various stages (eg. schedule generation, fleet assignment, aircraft routing, and crew pairing), with the decisions from one stage imposed upon the decision making process in subsequent stages. Whilst this approach greatly simplifies the solution process, it unfortunately fails to capture the many dependencies between the various stages, most notably between those of aircraft routing and crew pairing, and how these dependencies affect the propagation of delays through the flight network. As delays are commonly transferred between late running aircraft and crew, it is important that aircraft routing and crew pairing decisions are made together. The propagated delay may then be accurately estimated to minimize the overall propagated delay for the network and produce a robust solution for both aircraft and crew. In this paper we introduce a new approach to accurately calculate and minimize the cost of propagated delay, in a framework that integrates aircraft routing and crew pairing.

Key words: robust airline scheduling, delay propagation, airline schedule optimization.

1 Introduction

The airline scheduling problem involves the construction of timetables for an airline's major resources, namely aircraft and crew. Traditionally, this has been undertaken with a view towards maximizing an airline's overall profit, often with limited consideration given to the stability of such a schedule, or indeed its operational robustness. Such an approach has a tendency to generate schedules that are highly brittle, performing poorly in practice as delays propagate rapidly throughout the network. Lan *et al.* [12] states that in 2000, 30% of flight legs operated by a major US airline were delayed. In recent years, this has resulted in an ever increasing discrepancy between planned costs and realised operational costs. As aircraft networks continue to grow, this trend is set to continue with AhmadBeygi *et al.* [2] reporting that in 2006, it was estimated that the US airline industry experienced a total of 116.5 million minutes of delay; translating into a \$7.7 billion increase in operating costs. Such large discrepancies have prompted airline schedule planners to shift their focus from maximizing profit to maximizing robustness.

1.1 The airline scheduling problem

The airline scheduling problem in its entirety is very complex. The vast number of rules and regulations associated with airports, aircraft, and crew combined with the global expanse of air traffic networks, require the problem to be broken into manageable pieces to maintain some degree of tractability. Consequently, the traditional airline scheduling problem is typically decomposed into four stages, with the output of one stage used as the input for the subsequent stage(s). The very first stage is known as the *schedule generation* problem. In this step, an airline seeks to construct a schedule of flights where each flight is specified by an "origin, destination, departure date, time and duration" Weide *et al.* [22]. The

origin and destination of each flight leg (known as an OD pair), and additionally the frequency with which they are flown, are determined by the market demand for such pairs and availability of aircraft resources. The second stage, known as *fleet assignment* assigns a particular aircraft type (or fleet) to each flight leg, to appropriately match the size of the aircraft to the intended range (eg. long-haul vs domestic) and the expected number of passengers. Typically, the objective is to maximize profit via the minimization of operating expenses and number of spilled passengers. The third stage, known as *aircraft routing*, is performed separately for each specific fleet type to obtain a minimal cost assignment of aircraft to flights that ensures each flight is covered exactly once by exactly one aircraft. An aircraft routing is assigned to each aircraft, with each routing satisfying necessary maintenance requirements. Finally the last stage, known as *crew pairing*, is also performed separately for each fleet type, as crew typically may only fly on board a specific fleet. The objective of crew pairing is to find a minimal cost assignment of crew to flights. A set of crew pairings are constructed that satisfy union regulations (such as the 8-in-24 rule)¹, and ensure each flight is covered exactly once by exactly one crew group.

1.2 Integrated methods

The sequential solution approach, although easier to solve, may result in sub-optimal solutions as decisions fixed early in the process can limit flexibility in subsequent stages. This is primarily the result of the many interdependencies between the various stages. In a bid to more accurately model the airline scheduling problem, various authors have recently attempted to integrate two or more of these stages. Authors such as Desaulniers *et al.* [8] and Rexing *et al.* [16] have attempted to integrate the schedule generation process with fleet assignment via the discretization departure time windows for each flight, providing greater flexibility and a possibly more profitable solution. Klabjan *et al.* [11] and Lan *et al.* [12] perform a similar integration with crew pairing and aircraft routing respectively. Sandhu and Klabjan [18] note that the standard fleet assignment problem is solved with no consideration given to its impact on the quality of the crew pairing solution. To capture this dependency, the authors propose a model that integrates fleet assignment and crew pairing whilst maintaining the possibility of feasible aircraft routings by way of plane count constraints. Barnhart *et al.* [3] propose an approximate integrated model for fleet assignment and crew pairing.

Similar problems exist between the two stages of aircraft routing and crew pairing. As aircraft routing is typically fixed first, the solution requires the crew to change aircraft many times throughout the course of a duty period which may allow delays to propagate rapidly throughout the network. To address this, Cordeau *et al.* [7] integrate aircraft routing with crew pairing, using linking constraints to ensure that a crew does not swap aircraft if there is insufficient connection time. The problem is solved via Benders decomposition. Klabjan *et al.* [11] partially integrate aircraft routing with crew pairing. The authors solve the problem sequentially, adding plane count constraints to allow a feasible aircraft routing to be obtained. The authors also include time windows to allow greater flexibility within the crew pairing problem. Mercier *et al.* [13] improves upon the method of Cordeau *et al.* [7] through the introduction of so-called restricted connections. The authors allow restricted connections, but apply a penalty if both legs are covered in sequence by the same aircraft. The authors improve the speed of convergence by reversing the order in which the problems are solved so that the crew pairing is instead solved in the master problem. Mercier and Soumis [14] improve upon this further via the inclusion of time windows, in an attempt to integrate three aspects of the scheduling problem. Papadakos [15] integrates aircraft routing with crew pairing and re-timing (via time windows) and proposes an enhanced Benders decomposition, making use of a heuristic to circumvent the so-called tailing off effect associated with column generation to speed up convergence. Papadakos also notes that retaining the crew scheduling problem within the Benders subproblem leads to greater numerical efficiency.

1.3 Robust planning

As mentioned in the introduction, airline delays have increased dramatically in the last few years, resulting in ever increasing revenue losses for airlines. Up until recently, the primary focus of airline schedule planners was simply one of maximizing profit. However, since aircraft and crew are only profitable for an airline whilst they are in the air, the schedules generated by such an approach often contain very

¹The 8-in-24 rule is imposed by the FAA, and requires that crew be given additional rest should the total flying time of a pairing exceed 8 hours in a 24 hour period. See [4] for further details.

little slack between connecting flights for the absorption of delays. Consequently, in networks with a large number of connecting resources, delays can propagate very rapidly throughout the network. This in turn leads to significant recovery costs for an airline.

This ever increasing discrepancy between planned costs and realised costs has prompted airline schedule planners to shift their focus from maximizing profit to maximizing robustness within a schedule. In contrast to *airline recovery*, where the objective is to achieve the best course of action *after* an incident or delay has occurred, the focus of robust planning is to incorporate or establish an in-built level of robustness to unexpected occurrences. We outline a few approaches below. Ageeva [1] proposes a model that maximizes the number of times different aircraft routes ‘meet’. This provides an opportunity for aircraft to swap routes and return to their original route at some point in the future. This may prove beneficial if one aircraft is late and the other aircraft’s connection has a greater slack. Rosenberger *et al.* [17] propose a robust fleet assignment and aircraft routing model that produces a large number of short cycles with a low hub connectivity. A larger number of shorter cycles assists in preventing one single cancellation from causing a string of cancellations throughout the course of day. Schaefer *et al.* [19] solve a deterministic crew pairing problem where the costs of each pairing are estimated via a simulation tool known as SimAir. Yen and Birge [26] extend this approach, using a two-stage stochastic program to develop a robust crew pairing model. Their model identifies disruptions resulting from the first stage assignment decisions and their (non-linear) recourse model reflects interactions between long-range planning decisions and short-range operational results. Chebalov and Klabjan [6] propose a model that seeks to maximize the number of opportunities for crews to be swapped during operations. Smith *et al.* [20] propose a model where the number of different fleet types allowed to serve each airport is limited; this is called “imposing station purity”. Smith *et al.* demonstrate that this approach provides solutions that are robust for crew planning, maintenance planning and operations in general; however, this approach requires significant computational time.

Lan *et al.* [12] develop a robust aircraft routing model to minimize the expected propagated delay along aircraft routes. They use an approximate delay distribution to model the delay propagation along each string and use a branch and bound technique to solve their MIP. Lan *et al.* calculate propagated delay along individual strings when determining costs for the restricted master problem, but omit considerations of delay when solving the subproblem. The effect of connecting resources (such as crew and passengers) are not considered. Instead of estimating delay propagation, Wu [23] used a simulation model to calculate random ground operational delays and airborne delays in an airline network. Wu [23, 24] shows that delays are inherent in airline operations due to stochastic delay causes, e.g. passenger connections and late baggage loading. By adjusting flight times without changing aircraft routing, Wu [24] revealed that significant delay (cost) savings can be achieved via robust scheduling. Weide *et al.* [22] propose an integrated aircraft routing model for which the solution is obtained iteratively. The authors propose a non-robustness measure and initially solve the crew pairing problem without taking into account an aircraft routing solution. Their model then seeks to maximize the number of restricted connections contained in the aircraft solution that are also operated in the current crew pairing solution. Once this solution has been obtained, they minimize the number of restricted aircraft changes. This process continues iteratively, increasing the crew penalty at each iteration until the non-robustness measure cannot be improved further. The advantage of this approach is that the computational complexity is not increased as in other integrated models. AhmadBeygi *et al.* [2] make use of a propagation tree to minimize delay propagation due to flights and crew pairs in an existing routing and crew pairing solution, by re-timing flights so that the slack present in the network is re-allocated to where it is required most. Their approach is limited to retiming and both under and overestimates the delay propagation in certain cases.

1.4 Outline of this paper

Our aim is to improve upon the following shortcomings of AhmadBeygi *et al.* [2], Lan *et al.* [12] and Weide *et al.* [22]. Firstly, while Lan *et al.* correctly calculate propagated delay of aircraft strings in their master problem, the *selection* of these new columns is carried out more crudely: new columns are generated within the subproblem *without* considering the delay cost of the new column. The authors only make use of the dual variables from the master problem when determining the minimal cost column. Once a column has been generated they then calculate the propagated delay cost along the string and decide whether to add it to the restricted master problem. Furthermore, they ignore the effect of connecting

resources such as crew and passengers. Secondly, while AhmadBeygi *et al.* [2] consider (in a re-timing setting) the combined delay effects from crew and from aircraft, their approach imperfectly calculates how delays are propagated, resulting in possible under or overestimates of the true propagated delay. Their improvements are also limited to those achievable by retiming. Finally, Weide *et al.* [22] treat the interactions of crew and aircraft in an iterative fashion, optimizing a robustness measure, which is an *indirect* means of assessing the *true cost* due to total propagated delays of aircraft and crew.

We borrow the iterative theme of Weide [22], and alternate between aircraft routing and crew pairing master problems until no further reduction in total propagated delay is observed. The key ingredients of our approach are (i) the accurate calculation of the *combined effects* of propagation of delay along aircraft routing strings *and* crew pairing strings, (ii) the use of this information for both the calculation of the *cost* of columns and the dynamic *selection* of optimal columns, and (iii) the embedding of these calculations in an iterative scheme for the master routing and crew pairing problems.

In sections 2.1 and 2.2 we briefly outline standard column generation approaches to finding minimum cost aircraft routings and crew pairings, respectively. In section 2.3 we describe our approach for accurately calculating the propagated delay of routing and crewing strings and in section 2.4 we describe the setup of our pricing problems. Sections 3.1 and 3.2 describe our numerical approaches for solving the master and pricing problems, respectively. Computational results are presented in section 4 and we conclude with suggestions for future work in section 5.

2 The Integrated Problem Formulation

In this section we describe our formulation for the integrated aircraft routing and crew pairing problem; the objective is to minimize the total cost associated with propagated delay. We first outline the mathematical formulation of the aircraft routing and crew pairing problems individually and then discuss estimation of propagated delay and the corresponding pricing problem.

2.1 Aircraft Routing

The aircraft routing problem is performed separately for each specific fleet type. We seek a minimal cost assignment of aircraft to flights where each flight is covered exactly once by exactly one aircraft. The costs will represent the cost of the total delay incurred by the aircraft over a 24 hour period.

In the following routing model, we calculate a one day schedule and maintenance is performed at night. Maintenance feasible routings are represented as columns of an $m \times n_R$ binary matrix A^R , where m is the number of flights and n_R is the total number of feasible routings. The $(i, j)^{th}$ element of A^R takes the value 1 if flight i is contained in routing j and 0 otherwise. In practice there may be an extremely large number of feasible columns, so column generation is used to generate only the beneficial columns. For each flight (node) we assign a dollar cost per unit of delay arriving at that flight, and the cost c_j^R of column j is the sum of the costs of the delays along string j . The decision variable x_j^R takes the value 1 if routing j is included in the optimal solution and 0 otherwise. There is also an upper bound on the number of aircraft N . Thus we may state the aircraft routing problem as follows:

$$\begin{aligned} \text{minimize:} \quad & (\mathbf{c}^R)^T \mathbf{x}^R \\ \text{Subject to:} \quad & A^R \mathbf{x}^R = \mathbf{e} \\ & \sum_{i=1}^{n_R} x_i^R \leq N \\ & \mathbf{x}^R \in \{0, 1\}^{n_R} \end{aligned} \tag{1}$$

where \mathbf{e} is an m -dimensional column vector of 1s.

2.2 Crew pairing

The crew pairing problem is also performed separately for each fleet type, as crew typically may only fly on board a specific fleet. The objective of crew pairing is to find a minimal cost assignment of crew to flights. As in the routing problem, the costs will represent the dollar cost of the total propagated

delay incurred by the crew. A feasible set of crew pairings must satisfy union regulations (such as the 8-in-24 rule) and ensure each flight is covered exactly once by exactly one crew group. In the following crew pairing model, we assume a one day schedule where the crew are restricted to flying a total of less than eight hours in each pairing. However, we do not enforce any return to base conditions. As for the aircraft routing problem, the pairings may be represented as columns of an $m \times n_P$ matrix A^P , where m is the number of flights and n_P is the total number of feasible crew pairings. We use column generation to generate the most beneficial columns. The element c_j^P denotes the cost of column j and is defined as in the aircraft routing problem above. Thus, we may state the crew pairing problem as follows:

$$\begin{aligned} \text{minimize: } & (\mathbf{c}^P)^T \mathbf{x}^P \\ \text{Subject to: } & A^P \mathbf{x}^P = \mathbf{e} \\ & \mathbf{x}^P \in \{0, 1\}^{n_P} \end{aligned} \tag{2}$$

where \mathbf{e} is an m -dimensional column vector of 1s. There is typically no upper bound placed on the number of crews in the standard crew pairing problem.

2.3 Estimation of propagated delay

The calculation of total propagated delay along an aircraft string in an aircraft connection network or along a crew string in a crew connection network is non-trivial. The model of delay propagation we use for individual strings is based on a simplified version of Wu [23, 24] and is similar to the calculation of delay cost in individual strings used by Lan et al. We outline our modelling approach for calculation of propagated delay in the isolated routing and crewing networks before describing how to calculate propagated delay in a combined network in the next subsection.

Let $G = (\mathcal{N}, \mathcal{A})$ be a directed acyclic graph with a single source node s , and a single terminal node t . The source and terminal nodes are dummy nodes that link to both the morning and evening flights, respectively. In this graph, nodes correspond to flights and arcs correspond to possible feasible connections between flight nodes. For simplicity of exposition, we use the same connection network for both aircraft and crews, although one may use different arc sets if necessary.

Each connection $(i, j) \in \mathcal{A}$, will have associated with it two *primary delays*. The primary delay for aircraft connection (i, j) is denoted p_{ij}^R and is the sum of the expected en-route delay for flight i (estimated from historical data), and primary delays during aircraft turnaround operations, such as passenger connection delay, and ground handling delay. Note $p_{jt}^R = 0$ for all $(j, t) \in \mathcal{A}$. The primary delay for crew connection (i, j) is denoted p_{ij}^P and is the the sum of the expected en-route delay for flight i , and other crew related primary delays during aircraft turnaround time, such as late crew boarding and crewing procedures. Note $p_{jt}^P = 0$ for all $(j, t) \in \mathcal{A}$.

The flight schedule is the starting point for calculating *slack* for individual connections. The slack s_{ij} for a connection (i, j) is the difference between the scheduled arrival time of flight i and the scheduled departure time of flight j , minus the mean turn-around time for the relevant aircraft type under the specific ground handling procedure of the airline. All slacks $s_{si} = 0, (s, i) \in \mathcal{A}$, and $s_{jt} = 0, (j, t) \in \mathcal{A}$.

We now come to the *propagated delay* at node i , denoted d_i . We fix the initial delay at the source node $d_s = 0$ and inductively apply the formulae below to calculate propagated delay along a path in the aircraft connection network:

$$d_j^R = \max \{d_i^R - (s_{ij} - p_{ij}^R), 0\}, \quad j \neq s, \tag{3}$$

and in the crew connection network:

$$d_j^P = \max \{d_i^P - (s_{ij} - p_{ij}^P), 0\}, \quad j \neq s. \tag{4}$$

2.4 The pricing problems

In the previous section we saw how to calculate propagated delay along a path from the source node s . The delays along an aircraft string were only affected by aircraft delays in that string and not by delays due to connecting crew. Similarly, the delays when making a connection along a crew pairing string were affected by crew delays and enroute delays of the aircraft travelled on, but not by delays due to

the aircraft connected to. We now describe in more detail how we model the interaction between the routing and crewing problems and its effect on the pricing problems to be solved.

Firstly, we consider the effects of crew delays on the aircraft connection network. We assume that we are presented with a feasible set of crew strings and that propagated delays due to the crew have been calculated (to initialise the procedure, we will use (4) to calculate the d_i^P , $i \in \mathcal{N}$). To calculate the propagated delay along an aircraft string, *taking into account propagated delays from crew* we inductively apply:

$$d_j^R = \max \{d_i^R - (s_{ij} - p_{ij}^R), d_k^P - (s_{kj} - p_{kj}^P), 0\}, \quad j \neq s, \quad (5)$$

where the connection (i, j) is part of the aircraft string and the connection (k, j) is part of the crew string that includes flight j . Thus, if flight j uses the same aircraft as flight i and the same crew as flight k , the delay propagated to flight j is the maximum of the delays aircraft and crew (or zero, if both delays are negative); see Figure 1 for an example.

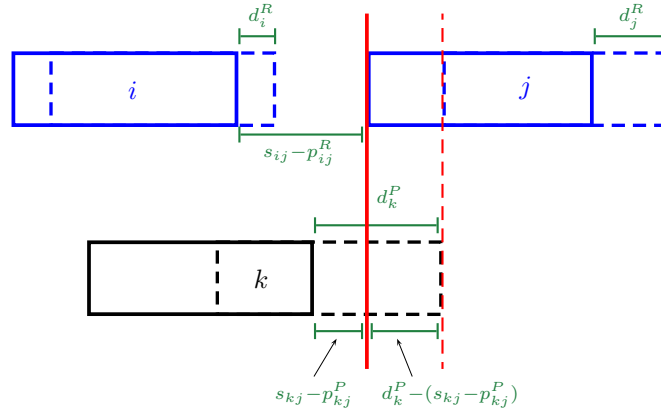


Figure 1: Illustration of the requirement of the maximum in equation (5). Aircraft and crew are denoted by blue and black boxes respectively. The bold red line denotes the scheduled departure time for flight j . Dashed lines represent the amount by which the aircraft and crew are delayed. Notice that although flight i is delayed, there is enough slack between flights i and j to absorb this delay. However, there is not enough slack between flights k and j for the crew on flight k to arrive in time for flight j . Thus, $d_k^P - (s_{kj} - p_{kj}^P) > 0$ and $d_j^R > 0$.

Secondly, we consider the effects of aircraft delays on the crew connection network. We assume that we are presented with a feasible set of aircraft strings and that propagated delays due to the aircraft have been calculated (to initialise the procedure, we will use (3) to calculate the d_i^R , $i \in \mathcal{N}$). As above, to calculate the propagated delay along a crew string, *taking into account propagated delays from aircraft* we inductively apply:

$$d_j^P = \max \{d_i^P - (s_{ij} - p_{ij}^P), d_k^R - (s_{kj} - p_{kj}^R), 0\}, \quad j \neq s, \quad (6)$$

where the connection (i, j) is part of the crew string and the connection (k, j) is part of the aircraft string that includes flight j .

We now describe the pricing problems for the routing and crewing master problems. When solving the routing subproblem the propagated routing delays d_i^R , $i \in \mathcal{N}$ will be calculated dynamically as part of the subproblem, using fixed pre-calculated propagated crewing delays d_i^P , $i \in \mathcal{N}$. When solving the crewing subproblem, the reverse is true; the propagated crewing delays d_i^P are dynamically calculated and the crewing delays d_i^R are pre-calculated and fixed.

Each node i possesses a weight $-w_i$, corresponding to the dual multiplier for constraint i in the master problem; we denote by $-w_i^R$ the weights from the routing master and by $-w_i^P$ the weights from the pairing master. We assume that for every unit of time an aircraft (resp. crew) is late at node i a dollar cost $a_i^R > 0$ (resp. $a_i^P > 0$) is incurred.

Thus, for the aircraft routing pricing problem we wish to find a path $\pi = \{s, i_1, i_2, \dots, t\}$ from s to t that minimizes

$$z^R = \min \left\{ \sum_{i \in \pi} (a_i^R d_i^R + w_i^R) : \pi \text{ is a path from } s \text{ to } t \right\}. \quad (7)$$

For the crew pairing pricing problem, we impose the additional upper limit H on the number of hours worked.

$$z^P = \min \left\{ \sum_{i \in \pi} (a_i^P d_i^P + w_i^P) : \begin{array}{l} \pi \text{ is a path from } s \text{ to } t, \\ \text{total hours worked} \leq H. \end{array} \right\}. \quad (8)$$

Upon obtaining a solution, the minimizing path (or string) is represented by a column \mathbf{x}_j , whose i^{th} element takes the value 1 if flight node i is in the path and 0 otherwise. A routing string is assigned a cost of

$$\begin{aligned} c_j^R &= z^R - \sum_{i \in \pi} w_i^R, \\ &= \sum_{i \in \pi} a_i^R d_i^R. \end{aligned} \quad (9)$$

and a crew pairing string is assigned a cost of

$$\begin{aligned} c_j^P &= z^P - \sum_{i \in \pi} w_i^P, \\ &= \sum_{i \in \pi} a_i^P d_i^P. \end{aligned} \quad (10)$$

In section 3.2 the z^R and z^P - minimizing paths are determined by a modified label setting algorithm that simultaneously calculates both the reduced cost of the path and the propagated delays.

3 Computational Approach

In this section we describe our iterative approach for handling the two master problems of aircraft routing and crew pairing, and our computational approach for solving the pricing problem.

3.1 Integrating Aircraft Routing and Crew Pairing.

We seek a minimal propagated delay cost solution to the integrated aircraft routing and crew pairing problem. It is well known (eg. [5, 22]) that both the aircraft routing and crew pairing problems are individually \mathcal{NP} -hard. To avoid any additional complexity, we adopt the theme of modelling the interactions between the aircraft and the crew in an iterative way from Weide *et al.* [22]. In the first version of our approach, we solve the integrated problem iteratively, beginning with the aircraft routing problem, linked to output from a crew pairing problem and then switching to the crew pairing problem linked to new output from the aircraft routing problem, and so on. We call this first approach Iterative Case A. We also study Iterative Case B, where the initial iteration begins with the crew pairing problem linked to output from an aircraft routing problem, and then proceeds to iterate as in Case A. The pricing problem solution approach is described in the next subsection.

We begin by introducing an updating algorithm that ensures stability of the propagated delays in the combined routing and crewing network.

Algorithm 3.1. (*Propagated Delay Evaluation*)

1. Using the strings from the incumbent routing and crew pairing solutions, update d_j^R inductively along each string in the incumbent routing solution using (5).
2. Using the strings from the incumbent routing and crew pairing solutions, update d_j^P inductively along each string in the incumbent crew pairing solution using (6).
3. If any d_j^R or d_j^P , $j \in \mathcal{N}$ changed in Step 1 or 2, then go to Step 1. Otherwise, STOP.

Algorithm 3.2. (*Iterative Case A*)

1. INITIALISATION:

- (a) Solve problems (1) and (2) respectively with the objective of determining the minimum number of aircraft N and the minimum number of crew required M , to cover all flights exactly once. We now have incumbent routing and crewing solutions.
- (b) For each arc $(i, j) \in \mathcal{A}$, assign expected primary delays p_{ij}^R and p_{ij}^P .
- (c) Set $d_k^P = 0$ for all $k \in \mathcal{N}$ and $d_s^R = 0$. Set an iteration counter $c = 0$.

2. MINIMUM DELAY AIRCRAFT ROUTING:

- (a) Apply Algorithm 3.1.
- (b) Assign costs to strings using (9). Solve problem (1) via column generation with the objective of minimizing the total delay cost to produce a new incumbent routing solution.

3. MINIMUM DELAY CREW PAIRING:

- (a) Apply Algorithm 3.1.
- (b) Assign costs to strings using (10). Solve problem (2) via column generation with the objective of minimizing the total delay cost to produce a new incumbent crew pairing solution.

4. If either the aircraft routing or crew pairing solution has changed, increment iteration counter $c \rightarrow c + 1$ and return to Step 2. Otherwise, STOP.

Algorithm 3.3. (*Iterative Case B*)

As for Algorithm 3.2, interchanging Steps 2 and 3, replacing Step 1(c) with:

- 1(c) Set $d_k^R = 0$ for all $k \in \mathcal{N}$ and $d_s^P = 0$. Set an iteration counter $c = 0$. and interchanging Steps 1 and 2 of Algorithm 3.1.

3.2 Solving the pricing problem

We describe the methodology to solve the pricing problem (7); the problem (8) requires straight-forward modifications described at the conclusion of this section. For each $i \in \mathcal{N}$ we are given a dual multiplier $-w_i^R$ ($-w_s^R = -w_t^R = 0$), a per unit delay cost a_i^R ($a_s^R = a_t^R = 0$), and propagated delays for crew pairings d_i^P . We wish to solve (7), where the d_i^R are calculated via (5). Because the delay d_i^R not a simple sum of delays along the path from s to i , the problem (7) is not easily cast as a minimum cost network flow. We propose a label setting algorithm, augmented by a notion of label dominance, modified from related problems in Desrochers and Soumis [9] and Dumitrescu *et al.* [10], that works efficiently in the cases tested.

Let π be a (full) path in G (an ordered collection of nodes $\{s, i_1, i_2, \dots, i_q, t\}$ in \mathcal{N} with $(s, i_1), (i_q, t) \in \mathcal{A}$ and $(i_\ell, i_{\ell+1}) \in \mathcal{A}$ for all $\ell = 1, \dots, q - 1$). For $i \in \pi$, let $\pi(i)$ denote the ordered collection of nodes in the path π truncated so that the final node in the list is i ; we will also call $\pi(i)$ a path. Define $W_{\pi(i)}^R = \sum_{j \in \pi(i)} w_j^R$. Denote by $d_{\pi(i)}^R$ the propagated expected routing delay at node i , computed along path $\pi(i)$ using (5), and define $A_{\pi(i)}^R = \sum_{j \in \pi(i)} a_j^R d_{\pi(j)}^R$.

In this terminology, we may rewrite (7) as

$$z^R = \min \left\{ A_{\pi(t)}^R + W_{\pi(t)}^R : \pi \text{ is a path from } s \text{ to } t \right\}. \quad (11)$$

Because of the nonlinear nature of the propagated routing delay formula (5), our labels must track both the accumulated cost $A_{\pi(i)}^R + W_{\pi(i)}^R$ at node i along path π , and the propagated delay $d_{\pi(i)}^R$. This motivates the following dominance conditions for labels.

Definition 3.4. (*Dominance condition*)

The pair (or label) $(A_{\pi(i)}^R + W_{\pi(i)}^R, d_{\pi(i)}^R)$ dominates $(A_{\eta(i)}^R + W_{\eta(i)}^R, d_{\eta(i)}^R)$ if

$$A_{\pi(i)}^R + W_{\pi(i)}^R \leq A_{\eta(i)}^R + W_{\eta(i)}^R \quad \text{and} \quad d_{\pi(i)}^R \leq d_{\eta(i)}^R$$

and the labels are not identical.

Lemma 3.5. Let ϖ be a path from j to k , where $(i, j) \in \mathcal{A}$. If $(A_{\pi(i)}^R + W_{\pi(i)}^R, d_{\pi(i)}^R)$ dominates $(A_{\eta(i)}^R + W_{\eta(i)}^R, d_{\eta(i)}^R)$, then $(A_{\{\pi(i), \varpi\}}^R + W_{\{\pi(i), \varpi\}}^R, d_{\{\pi(i), \varpi\}}^R)$ dominates $(A_{\{\eta(i), \varpi\}}^R + W_{\{\eta(i), \varpi\}}^R, d_{\{\eta(i), \varpi\}}^R)$.

Proof: We show that this is true if i connects to j by a single arc (the path ϖ consists of a single node $\{j\}$); the result then follows by induction. Recall we are given a fixed set of crew pairing strings. Let ξ denote the crew pairing string that includes flight node j and let k be the node in ξ preceding j . Thus,

$$\begin{aligned} d_{\{\pi(i), j\}}^R &= \max \left\{ d_{\pi(i)}^R - (s_{ij} - p_{ij}^R), d_{\xi(k)}^P - (s_{kj} - p_{kj}^P), 0 \right\}, \quad \text{and} \\ d_{\{\eta(i), j\}}^R &= \max \left\{ d_{\eta(i)}^R - (s_{ij} - p_{ij}^R), d_{\xi(k)}^P - (s_{kj} - p_{kj}^P), 0 \right\}. \end{aligned}$$

Since $d_{\pi(i)}^R \leq d_{\eta(i)}^R$, one has $d_{\{\pi(i), j\}}^R \leq d_{\{\eta(i), j\}}^R$.
Now

$$\begin{aligned} A_{\{\pi(i), j\}}^R + W_{\{\pi(i), j\}}^R &= A_{\pi(i)}^R + W_{\pi(i)}^R + a_j^R d_{\{\pi(i), j\}}^R + w_j^R \quad \text{and} \\ A_{\{\eta(i), j\}}^R + W_{\{\eta(i), j\}}^R &= A_{\eta(i)}^R + W_{\eta(i)}^R + a_j^R d_{\{\eta(i), j\}}^R + w_j^R, \end{aligned}$$

and we are done. ■

In particular, if ϖ terminates at t , the above lemma shows that $A_{\{\pi(i), \varpi\}}^R + W_{\{\pi(i), \varpi\}}^R \leq A_{\{\eta(i), \varpi\}}^R + W_{\{\eta(i), \varpi\}}^R$. In our labelling algorithm described below, we may therefore at each node only create labels for those paths which are not dominated by any other path at that node. We call such labels *efficient*.

Definition 3.6. A label $(A_{\pi(i)}^R + W_{\pi(i)}^R, d_{\pi(i)}^R)$ at node i is said to be efficient if it is not dominated by any other label at node i . A path $\pi(i)$ is said to be efficient if the label it corresponds to at node i is efficient.

We now describe the label setting algorithm we use to solve the problem (11). At a node $i \in \mathcal{N}$, the current collection of labels are denoted I_i and the current collection of treated labels we denote by M_i . Because the dominance condition does not allow identical labels at a node i , each label in I_i will correspond to a unique path (say $\pi(i)$) from s to i . For brevity, we will therefore denote individual elements of I_i and M_i as paths such as $\pi(i)$.

Algorithm 3.7. *Label Setting Algorithm for the Aircraft Routing Problem*

1. *Initialisation:*

Set $I_s = \{s\}$ and $I_i = \emptyset$ for all $i \in \mathcal{N} \setminus \{s\}$.

Set $M_i = \emptyset$ for each $i \in \mathcal{N}$.

2. *Selection of the label to be treated:*

if $\bigcup_{i \in \mathcal{N}} (I_i \setminus M_i) = \emptyset$ **then go to Step 4;** all efficient labels have been generated.

else choose $i \in \mathcal{N}$ and $\pi(i) \in I_i \setminus M_i$ so that $A_{\pi(i)}^R + W_{\pi(i)}^R$ is minimal.

3. *Treatment of label $(A_{\pi(i)}^R + W_{\pi(i)}^R, d_{\pi(i)}^R)$*

forall $(i, j) \in \mathcal{A}$

if $(A_{\{\pi(i), j\}}^R + W_{\{\pi(i), j\}}^R, d_{\{\pi(i), j\}}^R)$ is not dominated by $(A_{\eta(j)}^R + W_{\eta(j)}^R, d_{\eta(j)}^R)$ for any $\eta(j) \in I_j$

then

set $I_j = I_j \cup \{\pi(i), j\}$

end do

Set $M_i := M_i \cup \{\pi(i)\}$.

Go to Step 2.

4. *Return* $\arg \min_{\pi(t) \in I_t} A_{\pi(t)}^R + W_{\pi(t)}^R$.

Define $T_{\pi(i)} = \sum_{j \in \pi(i)} t_j$, where t_j is the scheduled time that crew work on flight j . We denote the allowed upper limit of continuous scheduled crew work time by H .

Algorithm 3.8. *Label Setting Algorithm for the Crew Pairing Problem*

As in Algorithm 3.7, replacing R superscripts by P superscripts throughout and replacing the **if** clause in Step 3 with:

if $T_{\{\pi(i),j\}} \leq H$ hours **and** $(A_{\{\pi(i),j\}}^R + W_{\{\pi(i),j\}}^P, d_{\{\pi(i),j\}}^P)$ is not dominated by $(A_{\eta(j)}^P + W_{\eta(j)}^P, d_{\eta(j)}^P)$ for any $\eta(j) \in I_j$ **then**
 set $I_j = I_j \cup \{\pi(i), j\}$

One could try to improve the efficiency of Algorithms 3.7 and 3.8, by for example using ideas from [10] for Algorithm 3.8. We found the Algorithms to be efficient on the instances tested and therefore have not explored further possible improvements.

4 Numerical Results

To evaluate the effectiveness of our proposed iterative approach, we apply Algorithm 3.2 to a one day schedule on a real airline network consisting of 53 flights and 126 feasible connections.

We determine that the minimum number of aircraft and crew pairs required to cover this network are 10 and 16, respectively, by solving (1) and (2). We use the corresponding aircraft routings and crew pairings to form our Base Case to which we apply our iterative integrated approach to reduce total propagated delay. We use 10 aircraft and 16 crew pairs in all instances and all algorithms tested.

The mean primary aircraft and crew pairing delays p_{ij}^R and p_{ij}^P are randomly sampled from four different probability distributions. In practice, primary aircraft and crew pairing delays rarely correspond to a specific distribution, but are rather a composite of several causes of delays with different individual distributions that may vary throughout different times of the day [21, 25]. It is often difficult to extract bias free, accurate historical data for the expected primary aircraft and crew delay over a specific connection. Thus, precise delay distributions (and their means) for all connections are very difficult to determine analytically. We therefore sample a set of delays and use the values obtained to represent a possible mean delay for each connection. To capture the asymmetric nature of the aircraft and crew delays, we sample from an exponential distribution $E(\lambda)$ with mean $1/\lambda$ in minutes and a truncated normal distribution (truncated to non-negative delays), denoted $tN(\mu, \sigma^2)$ with mean μ and variance σ , both in minutes. We test our new computational approach on 12 random instances: 3 instances from $E(1/5)$, 3 from $E(1/10)$, 3 from $tN(5, 100)$, and 3 from $tN(10, 25)$.

We study two simplified approaches in addition to our base case and proposed approach:

1. **Base (B):**

- Step 1 of Algorithm 3.2, followed by Algorithm 3.1.

2. **Routing and Crewing Solved Sequentially, Simple Delay (SSD):**

- Steps 1, 2, 3 of Algorithm 3.2, followed immediately by Algorithm 3.1. In Step 1 of Algorithm 3.1, (5) is replaced with $d_j^R = d_i^R - (s_{ij} - p_{ij}^R)$ and in Step 2 of Algorithm 3.1, (6) is replaced with $d_j^P = d_k^R - (s_{kj} - p_{kj}^R)$.

3. **Routing and Crewing Solved Sequentially, Propagated Delay (SPD):**

- Steps 1, 2, 3 of Algorithm 3.2, followed immediately by Algorithm 3.1.

4. **Routing and Crewing Integrated, Propagated Delay (IPD):**

- Algorithm 3.2.

The SPD approach will demonstrate the value of calculating the more accurate, nonlinear, *propagated delay* over the simpler, less accurate linear delay of the SSD approach. Our proposed IPD approach

will demonstrate the value of *integrating* routing and crewing, rather than simply performing them sequentially as in the SPD approach.

For each instance and each approach, we record in minutes the aircraft delay, crew delay, total delay, and improvement in total delay relative to the total delay incurred by the Base Case. In each approach we apply the evaluation Algorithm 3.1 to provide a consistent means of comparison between each of the approaches. Algorithm 3.1 takes either 1 or 2 iterations to complete for each of the 12 instances tested. Algorithm 3.2 takes only a small number of iterations to terminate; between 2 and 5 iterations for the 12 instances tested.

We remark that we evaluated Algorithm 3.3 on the same 12 instances and produced solutions that were universally inferior to Algorithm 3.2. This is not unexpected, as the routing strings are larger and less flexible than the crewing strings, and folklore suggests making decisions on less flexible items first often produces better results. The results for Algorithm 3.3 are thus not reported. Our numerical results for Algorithm 3.2 are tabulated below.

Exponential distribution with mean $\lambda = 5$.

Instance 1:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	207	841	1048	0.00
SSD	265	904	1169	-11.55
SPD	191	627	818	21.95
IPD	152	609	761	27.39

Instance 2:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	123	795	918	0.00
SSD	104	650	754	17.86
SPD	103	684	787	14.27
IPD	101	594	695	24.29

Instance 3:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	153	771	924	0.00
SSD	177	642	819	11.36
SPD	144	551	695	24.78
IPD	114	551	665	28.03

Exponential distribution with mean $\lambda = 10$.

Instance 4:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	704	1749	2453	0.00
SSD	633	1176	1809	26.25
SPD	610	1079	1689	31.15
IPD	540	1079	1619	34.00

Instance 5:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	1151	2133	3284	0.00
SSD	989	1604	2593	21.04
SPD	1031	1519	2550	22.35
IPD	867	1429	2296	30.09

Instance 6:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	848	2026	2874	0.00
SSD	691	1230	1921	33.16
SPD	727	1592	2319	19.31
IPD	494	1337	1831	36.29

Truncated Normal distribution with $\mu = 5$, $\sigma = 10$.

Instance 7:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	421	1389	1810	0.00
SSD	366	1044	1410	22.10
SPD	382	1038	1420	21.55
IPD	325	1040	1365	24.59

Instance 8:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	680	1577	2257	0.00
SSD	561	1333	1894	16.08
SPD	522	1116	1638	27.43
IPD	398	1156	1554	31.15

Instance 9:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	597	1718	2315	0.00
SSD	580	1287	1867	19.35
SPD	609	1308	1917	17.19
IPD	504	1318	1822	21.30

Truncated Normal distribution with $\mu = 10$, $\sigma = 5$.

Instance 10:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	352	1153	1505	0.00
SSD	324	932	1256	16.54
SPD	361	918	1279	15.02
IPD	290	928	1218	19.07

Instance 11:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	552	1509	2061	0.00
SSD	531	1129	1660	19.46
SPD	452	1139	1591	22.80
IPD	407	1139	1546	24.99

Instance 12:

Approach	Aircraft Delay (mins)	Crew Delay (mins)	Total Delay	% improv.
B	667	1795	2462	0.00
SSD	585	1418	2003	18.64
SPD	582	1460	2042	17.06
IPD	523	1393	1916	22.18

Table 1 details the relative improvements in delay between the algorithms SSD, SPD, and IPD for the 12 instances.

Table 1: Relative improvements of the algorithms SSD, SPD, and IPD.

Instance	$(SSD-SPD)/SSD*100\%$	$(SPD-IPD)/SPD*100\%$	$(SSD-IPD)/SSD*100\%$
1	30.03	6.97	34.90
2	-4.38	11.69	7.82
3	15.14	4.32	18.80
4	6.63	4.14	10.50
5	1.66	9.96	11.45
6	-20.72	21.04	4.69
7	-0.71	3.87	3.19
8	13.52	5.13	17.95
9	-2.68	4.96	2.41
10	-1.83	4.77	3.03
11	4.16	2.83	6.87
12	-1.95	6.17	4.34
Average	3.24	7.15	10.50

5 Discussion and Conclusions

Our iterative integrated methodology for minimizing propagated delay in a combined routing and crewing network has clear advantages over approaches that do not explicitly calculate propagated delay or fail to properly integrate routing and crewing.

- The value of integrating routing and crewing, rather than sequentially minimizing propagated delay in routing strings, then minimizing propagated delay in crew strings is clear from a comparison of IPD and SPD delays in our 12 instances. There is universal improvement over all instances; on average our IPD approach improves by 7.2% over the SPD approach.
- For the two sequential approaches tested, the value of accurately calculating propagated delay, rather than using a simpler additive delay, is less clear, with only half of the SPD delays improving over the SSD delays. On average over the 12 instances, however, the SPD approach improves over SSD by 3.2%.
- Finally, integrating routing and crew delays and accurately calculating the propagated delays (our IPD approach) is a clear and universal improvement over SSD with an average improvement of 10.5%.

In this proof of concept work, we have limited our study to minimizing expected propagated delay, however, our methodology allows other extensions to mitigate delay related risk. For example, it is straightforward to limit the maximum expected propagated delay of any single flight. In Algorithm 3.7, one may disallow the creation of a path with an unacceptably high single flight delay cost in the same way that crew strings of duration greater than H hours are disallowed in Algorithm 3.8. Similarly, it is easy to limit the total delay cost of either a routing or crew string.

Our new integrated framework is in principle extendable to a third aspect, such as delays due to passengers. Future work will explore this exciting possibility.

6 Acknowledgements

We thank Irina Dumitrescu for helpful discussions on labelling algorithms. MD acknowledges support from the ARC Centre of Excellence for Mathematics and Statistics of Complex Systems (MASCOS) and an Australian Postgraduate Award (APA). GF is partially supported by MASCOS.

References

- [1] Y. Ageeva. Approaches to Incorporating Robustness into Airline Scheduling. Master's thesis, Massachusetts Institute of Technology, 2000.
- [2] S. AhmadBeygi, A. Cohn, and M. Lapp. Decreasing Airline Delay Propagation by Re-Allocating Scheduled Slack. In *Proceedings of the Industry Studies Conference*, 2008.
- [3] C. Barnhart, N. L. Boland, L. W. Clarke, E. L. Johnson, G. L. Nemhauser, and R. G. Sheno. Flight string models for aircraft fleeting and routing. *Transportation Science*, 32(3):208–220, 1998.
- [4] C. Barnhart, A. M. Cohn, D. Klabjan, G. L. Nemhauser, and P. Vance. Airline crew scheduling. In R. W. Hall, editor, *Handbook of Transportation Science*, pages 517–560. Kluwer, 2003.
- [5] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance. Branch-and-Price: Column Generation for Solving Huge Integer Programs. *Operations Research*, 46(3):316–329, 1998.
- [6] S. Chebalov and D. Klabjan. Robust Airline Crew Pairing: Move Up Crews. *Transportation Science*, 40(3):300–312, 2006.
- [7] J. F. Cordeau, G. Stojkovic, F. Soumis, and J. Desrosiers. Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling. *Transportation Science*, 35(4):375–388, 2001.
- [8] G. Desaulniers, J. Desrosiers, Y. Dumas, M. M. Solomon, and F. Soumis. Daily Aircraft Routing and Scheduling. *Management Science*, 43(6):841–855, 1997.
- [9] M. Desrochers and F. Soumis. A Generalized Permanent Labelling Algorithm for the Shortest Path Problem with Time Windows. *Infor*, 26(3):191–212, 1988.
- [10] I. Dumitrescu and N. Boland. Improved preprocessing, labeling and scaling algorithms for the weight-constrained shortest path problem. *Networks*, 42(3):135–153, 2003.
- [11] D. Klabjan, E. L. Johnson, G. L. Nemhauser, E. Gelman, and S. Ramaswamy. Airline Crew Scheduling with Time Windows and Plane-Count Constraints. *Transportation Science*, 36(3):337–348, 2002.
- [12] S. Lan, J. Clarke, and C. Barnhart. Planning for Robust Airline Operations Optimizing Aircraft Routings and Flight Departure Times to Minimize Passenger Disruptions. *Transportation Science*, 40(1):15–28, 2006.
- [13] A. Mercier, J. F. Cordeau, and F. Soumis. A Computational Study of Benders Decomposition for the Integrated Aircraft Routing and Crew Scheduling Problem. *Computational Operations Research*, 32(6):1451–1476, 2005.
- [14] A. Mercier and F. Soumis. An Integrated Aircraft Routing, Crew Scheduling and Flight Retiming Model. *Computers and Operations Research*, 34:2251–2265, 2007.
- [15] N. Papadacos. Integrated Airline Scheduling. *Computers and Operations Research*, 36(1):176–195, 2009.
- [16] B. Rexing, C. Barnhart, T. Kniker, A. Jarrah, and N. Krishnamurthy. Airline Fleet Assignment with Time Windows. *Transportation Science*, 34(1):1–20, 2000.
- [17] J. Rosenberger, E. Johnson, and G. Nemhauser. A Robust Fleet Assignment Model with Hub Isolation and Short Cycles. *Transportation Science*, 38(3):357–368, 2004.
- [18] R. Sandhu and D. Klabjan. Integrated Airline Fleetting and Crew-Pairing Decisions. *Operations Research*, 55(3):439–456, 2007.
- [19] A. Schaefer, E. L. Johnson, A. J. Kleywegt, and G. L. Nemhauser. Airline Crew Scheduling Under Uncertainty. *Transportation Science*, 39(3):340–348, 2005.

- [20] B. Smith and E. L. Johnson. Robust Airline Fleet Assignment: Imposing station purity using station decomposition. *Transportation Science*, 40(4):497–516, 2006.
- [21] Y. Tu, M. O. Ball, and W. S. Jank. Estimating Flight Departure Delay Distributions: A Statistical Approach With Long-Term Trend and Short-Term Pattern. *Journal of the American Statistical Association*, 103:112–125, 2008.
- [22] O. Weide, D. Ryan, and M. Ehrgott. An Iterative Approach to Robust and Integrated Aircraft Routing and Crew Scheduling. Elsevier, 2009. doi:10.1016/j.cor.2009.03.024.
- [23] C. L. Wu. Inherent Delays and Operational Reliability of Airline Schedules. *Air Transport Management*, 11:273–282, 2005.
- [24] C. L. Wu. Improving Airline Robustness and Operational Reliability by Sequential Optimisation Algorithms. *Network and Spatial Economics*, 6:235–251, 2006.
- [25] C. L. Wu. *Airline Operations and Delay Management - Insights from Airline Economics, Networks and Strategic Schedule Planning*. Ashgate, 2010. IN PRESS.
- [26] J. W. Yen and J. R. Birge. A Stochastic Programming Approach to the Airline Crew Scheduling Problem. *Transportation Science*, 40(1):3–14, 2006.