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

For reasons 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.
(To appear in Transportation Science).

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. The Bureau of Transportation Statistics [18] states that in 2009, approximately 23% of flight legs operated by a major US airline were delayed – with late arrivals and cancellations combined accounting for more than 7.5% of this delay. 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 expected profits under uncertainty, by including various types of costs arising from unplanned events.

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.* [25]. 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.* [19] 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.* [13] and Lan *et al.* [14] perform a similar integration with crew pairing and aircraft routing respectively. Sandhu and Klabjan [21] 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.* [13] 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.* [15] 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 [16] improve upon this further via the inclusion of time windows, in an attempt to integrate three aspects of the scheduling problem. Papadakos [17] 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] The 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.

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 [2]. For example, the average delays per flight in Europe increased dramatically from 12 minutes in 2006 to 28 minutes in 2009 [11]. 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 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 expected profits that include some costs due to unforeseen events. 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.* [20] 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.* [22] solve a deterministic crew pairing problem where the costs of each pairing are estimated via a simulation tool known as SimAir. Yen and Birge [29] 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.* [23] 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.* [14] 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 [26] used a simulation model to calculate random ground operational delays and airborne delays in an airline network. Wu [26, 27] 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 [27] revealed that significant delay (cost) savings can be achieved via robust scheduling. Weide *et al.* [25] 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.* [14] and Weide *et al.* [25]. 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.* [25] 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. The model in [25] attempts to keep aircraft and crew together over restricted connections, to try to minimise the number of restricted aircraft changes. Although [25] takes into account the connection time, penalising shorter restricted aircraft changes more severely, the Weide *et al.* model penalties are time-of-day independent, independent of historical information for the network, and does not quantitatively assess the propagated delay from the interactive connectivity of the routing and crewing networks. For example, there may be relatively predictable large primary delays over certain connections or at certain times of the day, or the effects of delays for some connections are much worse in a propagated sense than for other connections, depending on the interactive network topology. Our approach explicitly utilises time-of-day historical primary delays and explicitly calculates and minimises the downstream effect of delay in the combined routing and crewing network. Solutions developed from our approach may (for example) mismatch aircraft and crew on a restricted connection if later connections have ample slack to absorb delays. This mismatch may free up the possibility to match crew and aircraft on a critical connection that has tight connections further downstream. We provide a quantitative comparison of our approach and the approach of [25] in Section 4.

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 and (ii) the use of this information for both the calculation of the *cost* of columns and the dynamic *selection* of optimal columns.

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 2.5 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. We concentrate solely on costs due to delays with the understanding that in practice, the additional costs due to unplanned delays can form part of an overall model of cost for the airline. We thus view our proposed methodology as a potential add-on to existing connection-based optimisation models to better reflect planned costs under uncertainty.

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 where each aircraft begins and ends its day at a maintenance base. 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. The airline from which we source our data uses both pay-and-credits (for cabin crew) and flying hours (for pilots) as crew payment bases. For the purposes of this paper we use the flying-hour based crew costing model, which simplifies our crew costing model. 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 (8-in-24 rule) and ensure that at the end of its duty, each crew pairing returns to the crew base at which it started. This modified 8-in-24 assumption for a one-day schedule simplifies our crew pairing model. One could relax this assumption and expand the schedule to one week during implementation. 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:

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

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 [26, 27] 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 so, 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^R_{ij} 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 $p^R_{jt} = 0$ for all $(j, t) \in \mathcal{A}$. The primary delay for crew connection (i, j) is denoted p^P_{ij} 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. Enroute delays and turnaround delays occur for a variety of reasons such as weather conditions, air traffic flow management, passenger delays, equipment failure, and so on. These delays and their causes are documented by airlines by using the IATA delay coding system or its in-house variant [12]. Note $p^P_{jt} = 0$ 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. The value of the mean turn-around time is determined by the standard aircraft ground operating procedures of a specific fleet by an airline. Airlines design aircraft turn-around time based on the mean turn-around time and buffer allowance. For simplicity we have used the same turn-around time for all connections, as all aircraft belong to the same fleet and operate on a domestic network. It is however, straightforward to specify specific turn-around times for individual connections should this be required for an alternative network. All slacks $s_{so,i} = 0, (so, i) \in \mathcal{A}$, 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 $d_{so} = 0$ and inductively apply the formulae below to calculate propagated delay along a path in the aircraft connection network:

$$d^R_j = \max\left\{d^R_i - (s_{ij} - p^R_{ij}), 0\right\}, \quad j \neq so, \tag{3}$$

and in the crew connection network:

$$d^P_j = \max\left\{d^P_i - (s_{ij} - p^P_{ij}), 0\right\}, \quad j \neq so. \tag{4}$$

2.4 Estimation of combined propagated delay

In the previous section we saw how to calculate propagated delay along a path from the source node so. The delays along an aircraft string were only affected by aircraft delays in that string and not by delays due to connecting crew. Similarly, delays along a crew pairing were only affected by crew delays in that string and not delays due to connecting aircraft. 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^P_i, $i \in \mathcal{N}$). To calculate the propagated delay along an aircraft string, *taking into account propagated delays from crew* we inductively apply:

$$d^R_j = \max\left\{d^R_i - (s_{ij} - p^R_{ij}), d^P_k - (s_{kj} - p^P_{kj}), 0\right\}, \quad j \neq so, \tag{5}$$

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 of the aircraft and crew (or zero, if both delays are negative); see Figure 1 for an example.

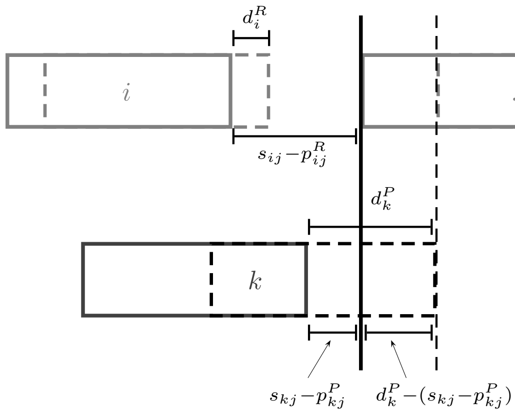

Figure 1: Illustration of the requirement of the maximum in equation (5). Aircraft and crew are denoted by blue and black boxes respectively. The bold red line denotes the scheduled departure time for flight j. Dashed lines represent the amount by which the aircraft and crew are delayed. Notice that 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:

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

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.

2.5 The pricing problems

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. These costs are combinations of costs associated with excess fuel consumption, overtime pay for crew, and costs associated with reaccommodating misconnecting passengers [2].

Finally, for the route pricing (resp. crew pricing) we add approximate reduced cost terms to represent the impact of inserting a particular route (resp. crew string) on overall crew delay (resp. routing delay). We describe these ideas for the routing pricing problem; the approach for the crew pricing problem is completely analogous. Consider node j and suppose that our incumbent routing solution has a connection (ℓ, j) and our incumbent crewing solution has a connection (k, j). The combined propagated routing and crewing delays at node j are given by

$$d_j^R = \max\left\{d_\ell^R - (s_{\ell j} - p_{\ell j}^R),\ d_k^P - (s_{kj} - p_{kj}^P),\ 0\right\}, \tag{7}$$

$$d_j^P = \max\left\{d_k^P - (s_{kj} - p_{kj}^P),\ d_\ell^R - (s_{\ell j} - p_{\ell j}^R),\ 0\right\}, \tag{8}$$

Suppose that in the current routing pricing problem we consider replacing the aircraft connection (ℓ, j) with (i, j). We calculate d_j^R along the routing string being constructed using (5). If this potential replacement string is inserted into master problem basis, there will be an impact on the crew delays. Using (6), at node j, the new (locally calculated) crew delay is given by

$$\tilde{d}_{j;i}^P = \max\left\{d_k^P - (s_{kj} - p_{kj}^P),\ d_i^R - (s_{ij} - p_{ij}^R),\ 0\right\}; \tag{9}$$

where the tilde is used to denote a temporary calculation local to node j, using the information that i is the prior node. We will use $a_j^P(\tilde{d}_{j;i}^P - d_j^P)$ as an estimate of the reduced cost for crew delay attributable to node j, for the routing string under construction.

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

$$z^R = \min\left\{\sum_{i\in\pi}\left(a_i^R d_i^R + w_i^R + a_i^P(\tilde{d}_{i;\pi^-(i)}^P - d_i^P)\right)\ :\ \pi \text{ is a path from } so \text{ to } t\right\}, \tag{10}$$

where $\pi^-(i)$ denotes the node prior to i in path π and with the further restriction that the path π begins and ends at a maintenance base.

For the crew pricing problem, a completely analogous procedure is used to construct the reduced cost estimate $a_j^R(\tilde{d}_{j;\pi^-(j)}^R - d_j^R)$ for the routing delay, attributable to node j, from the crew string under construction.

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

$$z^P = \min\left\{\sum_{i\in\pi}\left(a_i^P d_i^P + w_i^P + a_i^R(\tilde{d}_{i;\pi^-(i)}^R - d_i^R)\right)\ :\ \begin{array}{c}\pi \text{ is a path from } so \text{ to } t,\\ \text{total hours worked} \le H.\end{array}\right\}, \tag{11}$$

with the further restriction that the path π begins and ends at the same crew base.

Upon obtaining a solution to (10) (resp. (11)), the minimizing path (or string) forms a column A_j of the matrix A^R (resp. A^P). A routing string is assigned a cost of

$$\begin{aligned} c_j^R &= z^R - \sum_{i\in\pi} w_i^R, \\ &= \sum_{i\in\pi}\left(a_i^R d_i^R + a_i^P(\tilde{d}_{i;\pi^-(i)}^P - d_i^P)\right). \end{aligned} \tag{12}$$

and a crew pairing string is assigned a cost of

$$\begin{aligned} c_j^P &= z^P - \sum_{i\in\pi} w_i^P, \\ &= \sum_{i\in\pi}\left(a_i^P d_i^P + a_i^R(\tilde{d}_{i;\pi^-(i)}^R - d_i^R)\right). \end{aligned} \tag{13}$$

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, 25]) 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.* [25]. 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. This approach is not exact, however we have carefully modelled the crew and aircraft delay interactions and expect to obtain solutions of good quality. In Section 4 we demonstrate that we achieve significant improvements over standard approaches and our solutions also compare well against a rigorous lower bound. 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. *Perform a topological sorting of the flight nodes so that the flights are sorted from earliest to latest.*
2. *Using the strings from the incumbent routing and crew pairing solutions, update d_j^R and d_j^P together by inductively applying equations (5) and (6); moving strictly forwards in time throughout the day.*

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}^R.*
   (c) *Set $d_k^P = 0$, $d_k^R = 0$ for all $k \in \mathcal{N}$ and $d_{so}^R = 0$, $d_{so}^P = 0$. Set an iteration counter $c = 0$.*
2. Minimum Delay Aircraft Routing:
   (a) *Apply Algorithm 3.1.*
   (b) *Assign delay costs to strings using (12). 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 delay costs to strings using (13). 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 $c \to c+1$ and return to Step 2. Otherwise, goto Step 5.*
5. *Return $\sum_{n=1}^{N} \sum_{i \in \pi_n^R} a_i^R d_i^R + \sum_{m=1}^{M} \sum_{i \in \pi_m^P} a_i^P d_i^P$, where π_n^R is the routing string for the n^{th} aircraft, $n = 1, \ldots, N$ and π_m^P is the crew pairing string for the m^{th} crew, $m = 1, \ldots, M$.*

Algorithm 3.3. *(Iterative Case B)*
As for Algorithm 3.2, interchanging Steps 2 and 3.

3.2 Solving the pricing problem

We describe the methodology to solve the pricing problem (10); the problem (11) requires straightforward modifications described at the conclusion of this section. For each $i \in \mathcal{N}$, we are given a dual multiplier $-w_i^R$ ($-w_{so}^R = -w_t^R = 0$), a per unit delay cost a_i^R ($a_{so}^R = a_t^R = 0$), and propagated delays for crew pairings d_i^P. We wish to solve (10), where the d_i^R are calculated via (5). Because the delay d_i^R is not a simple sum of delays along the path from so to i, the problem (10) 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 $\{so, i_1, i_2, \ldots, i_q, t\}$ in $\mathcal{N}$ with $(so, i_1), (i_q, t) \in \mathcal{A}$ and $(i_\ell, i_{\ell+1}) \in \mathcal{A}$ for all $\ell = 1, \ldots, q-1$). 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 $W_{\pi(i)}^R = \sum_{j \in \pi(i)} \left(w_j^R + a_j^P (\tilde{d}_{j;\pi^-(j)}^R - d_j^R) \right)$. 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 (10) as

$$z^R = \min \left\{ A_{\pi(t)}^R + W_{\pi(t)}^R \ : \ \pi \text{ is a path from } so \text{ to } t \right\}. \tag{14}$$

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,

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

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

Now

$$A_{\{\pi(i),j\}}^R + W_{\{\pi(i),j\}}^R = A_{\pi(i)}^R + W_{\pi(i)}^R + a_j^R d_{\{\pi(i),j\}}^R + w_j^R + a_j^P (\tilde{d}_{j;i}^R - d_j^R) \quad \text{and}$$
$$A_{\{\eta(i),j\}}^R + W_{\{\eta(i),j\}}^R = A_{\eta(i)}^R + W_{\eta(i)}^R + a_j^R d_{\{\eta(i),j\}}^R + w_j^R + a_j^P (\tilde{d}_{j;i}^R - d_j^R),$$

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 (14). 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 *so* 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 $I_{so} = \{so\}$ *and* $I_i = \emptyset$ *for all* $i \in \mathcal{N} \backslash \{so\}$.
   Set $M_i = \emptyset$ *for each* $i \in \mathcal{N}$.

2. *Selection of the label to be treated:*
   if $\bigcup_{i \in \mathcal{N}} (I_i \backslash M_i) = \emptyset$ **then go to** *Step 4; all efficient labels have been generated.*
   else *choose* $i \in \mathcal{N}$ *and* $\pi(i) \in I_i \backslash M_i$ *so that* $A^R_{\pi(i)} + W^R_{\pi(i)}$ *is minimal.*

3. *Treatment of label* $(A^R_{\pi(i)} + W^R_{\pi(i)}, d^R_{\pi(i)})$
   forall $(i, j) \in \mathcal{A}$
   if $(A^R_{\{\pi(i),j\}} + W^R_{\{\pi(i),j\}}, d^R_{\{\pi(i),j\}})$ *is not dominated by* $(A^R_{\eta(j)} + W^R_{\eta(j)}, d^R_{\eta(j)})$ *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* $\arg\min_{\pi(t) \in I_t} A^R_{\pi(t)} + W^R_{\pi(t)}$.

We now describe the modifications required to solve the corresponding problem for the crew. 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. Equation (11) can be written as

$$z^P = \min \left\{ A^P_{\pi(t)} + W^P_{\pi(t)} \ : \ \pi \text{ is a path from } so \text{ to } t,\, T_{\pi(t)} \leq H \right\}. \tag{15}$$

Definition 3.8. *(Dominance condition)*
The pair (or label) $(A^R_{\pi(i)} + W^R_{\pi(i)},\ d^R_{\pi(i)},\ T_{\pi(i)})$ *dominates* $(A^R_{\eta(i)} + W^R_{\eta(i)},\ d^R_{\eta(i)},\ T_{\eta(i)})$ *if*

$$A^R_{\pi(i)} + W^R_{\pi(i)} \leq A^R_{\eta(i)} + W^R_{\eta(i)} \quad \text{and} \quad d^R_{\pi(i)} \leq d^R_{\eta(i)} \quad \text{and} \quad T_{\pi(i)} \leq T_{\eta(i)}$$

and the labels are not identical.

In Algorithm 3.9 we do not propagate paths to a node i if $T_{\pi(i)} > H$.

Algorithm 3.9. *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** $(A^P_{\{\pi(i),j\}} + W^P_{\{\pi(i),j\}}, d^P_{\{\pi(i),j\}}, T_{\{\pi(i),j\}})$ *is not dominated by* $(A^P_{\eta(j)} + W^P_{\eta(j)}, d^P_{\eta(j)}, T_{\eta(j)})$ *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 3.9, by for example using ideas from [10] for Algorithm 3.9. 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 54 flights and 128 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). For simplicity we assume that all aircraft, crew and connections incur similar operating costs, and thus the minimum number of aircraft and crew pairs solution represents a cost minimization without regard for costs due to unforeseen delays. 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 [24, 28]. 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 use unit costs per unit delay for all connections.

We study two simplified approaches (SSD) and (SSP) in addition to our base case (B) and proposed approach (IPD). We also compare our results with the method of [25] (W) as well as a proposed improvement to the method of [25] (WI):

1. **Base (B):**

   - Step 1 of Algorithm 3.2, followed by Algorithm 3.1 and Step 5 of Algorithm 3.2.

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

   - Steps 1, 2, 3 of Algorithm 3.2, followed immediately by Algorithm 3.1 and Step 5 of Algorithm 3.2. In Algorithm 3.1, (5) is replaced with $d_j^R = d_i^R - (s_{ij} - p_{ij}^R)$ and (6) is replaced with $d_j^P = d_k^R - (s_{kj} - p_{kj}^R)$. In Algorithm 3.2, (12) is replaced with $c_j^R = \sum_{i\in\pi} a_i^R d_i^R$ and (13) is replaced with $c_j^P = \sum_{i\in\pi} a_i^P d_i^P$.

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

   - Steps 1, 2, 3 of Algorithm 3.2, followed immediately by Algorithm 3.1 and Step 5 of Algorithm 3.2.

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

   - Algorithm 3.2.

5. **The Algorithm of Weide *et al.* [25] (W)**
   - The algorithm as described in Weide *et al.*. In the absence of cost-differentation for different crew pairings, we set the crew pairing cost to zero.

6. **An Improved version of the Algorithm of Weide *et al.* [25] (WI)**
   - The algorithm W, with an attempt to incorporate a "time-of-day" aspect based on expected primary delay. Compute restricted connections using the scheduled slack minus the expected primary delay, instead of scheduled slack.

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. The SPD approach may be viewed as an improvement over Lan *et al.* [14] because we use the correct calculation of propagated delay in column selection and also model interaction of aircraft and crew (see discussion in Section 1.4). The IPD approach is an improvement over AhmadBeygi *et al.* [2] as we correctly calculate the combined propagated delay due to aircraft and crew; moreover, we develop routing and crewing connections, rather than retiming existing connections. We also view IPD as an improvement over Weide *et al.* [25] as our objective is in terms of a dollar cost, which can be easily added to other operating cost terms in a more sophisticated cost model. We compare our IPD approach with the model of Weide *et al.* W and also with the "improved" model WI.

For each instance and each of the approaches SSD, SPD, and IPD, 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 3.2 takes between 3 and 16 iterations for the 12 instances tested, as indicated in the tables below.

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.

The IP was always solved at the root node by column generation and did not require any further branching. As the network consisted of 54 flights, the master problem consisted of 54 set partitioning constraints for both the aircraft routing and crew pairing problems. Approximately 200 columns were generated in an aircraft routing iteration and approximately 120 in a crew pairing iteration.

We also solved (1) and (2) separately to minimize the individual propagated delay due to aircraft and crew, respectively. These values are tabulated below, along with their sum, which represents a rigorous lower bound. This lower bound is unlikely to be sharp as it completely ignores the additional delays due to the combination of aircraft and crew delay; in some instances this combined effect can be substantial. In most instances our IPD solution is close to this lower bound; given the lack of sharpness of this bound, the IPD solutions appear to be of high quality. When running the algorithms W and WI, we found that as our network consists of many restricted connections, we could not achieve a non-robustness measure (NRM) of zero; but rather terminated when the NRM could not be improved further, as stipulated in [25]. For each instance, there were 9 restricted aircraft changes in the final solution; 8 of these may be classified as "less severe", as the sit time exceeded the minimum sit time by more than 15 minutes.

Our numerical results for Algorithm 3.2 are tabulated below. Individual results are given for each instance, followed by a summary in Table 1, detailing the relative improvements in delay between the algorithms SSD, SPD, IPD, W, and WI. All experiments were done with CPLEX12.1 on a 2.4GHz PC with 4GB RAM.

Exponential distribution with mean $\lambda = 5$.

Instance 1:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	214	316	530	–	9.17
SSD	155	229	384	27.55	21.53
SPD	146	229	375	29.25	28.41
IPD (3 iter.)	132	229	361	31.89	47.19
Lower Bound	106	210	316	–	–
W (10 iter.)	143	236	379	–	12.75
WI (8 iter.)	138	232	370	–	12.48

Instance 2:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	367	395	762	–	9.10
SSD	326	394	720	5.51	23.56
SPD	326	379	705	7.48	31.20
IPD (8 iter.)	321	347	668	12.34	68.01
Lower Bound	177	335	512	–	–
W (10 iter.)	350	390	740	–	12.75
WI (9 iter.)	349	388	737	–	10.77

Instance 3:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	158	316	474	–	10.15
SSD	164	295	459	3.16	23.41
SPD	160	297	457	3.59	28.55
IPD (7 iter.)	116	297	413	12.87	63.45
Lower Bound	104	275	379	–	–
W (10. iter)	141	316	457	–	12.75
WI (10 iter.)	126	315	441	–	15.02

Exponential distribution with mean $\lambda = 10$.

Instance 4:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	341	544	885	–	8.12
SSD	312	501	813	8.14	24.21
SPD	267	478	745	15.82	29.50
IPD (4 iter.)	241	471	712	19.55	72.26
Lower Bound	185	468	653	–	–
W (10 iter.)	312	501	813	–	12.75
WI (10 iter.)	304	485	789	–	16.00

Instance 5:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	999	1114	2113	–	15.53
SSD	826	1216	2042	3.36	26.22
SPD	856	1039	1895	10.32	28.16
IPD (16 iter.)	825	879	1704	19.36	214.19
Lower Bound	590	879	1469	–	–
W (10 iter.)	895	1042	1937	–	12.75
WI (8 iter.)	890	1027	1917	–	11.78

Instance 6:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	1217	1846	3063	–	14.51
SSD	1117	1653	2770	9.57	22.34
SPD	1108	1516	2624	14.33	23.19
IPD (4 iter.)	1032	1500	2532	17.34	92.35
Lower Bound	994	1456	2450	–	–
W (10 iter.)	1070	1589	2659	–	12.75
WI (10 iter.)	1053	1573	2626	–	11.33

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

Instance 7:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	438	665	1103	–	14.19
SSD	465	598	1063	3.63	22.10
SPD	441	598	1039	5.80	25.46
IPD (4 iter.)	387	573	960	12.96	39.44
Lower Bound	260	434	694	–	–
W (10 iter.)	425	591	1016	–	12.75
WI (8 iter.)	416	582	998	–	10.80

Instance 8:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	536	650	1186	–	13.31
SSD	503	689	1192	−0.51	24.75
SPD	503	652	1155	2.61	25.91
IPD (7 iter.)	505	571	1076	9.27	168.74
Lower Bound	481	562	1043	–	–
W (10 iter.)	526	647	1173	–	12.75
WI (9 iter.)	524	645	1169	–	12.56

Instance 9:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	274	562	836	–	15.11
SSD	269	408	677	19.02	27.89
SPD	260	434	694	16.99	28.61
IPD (6 iter.)	227	408	635	24.04	57.98
Lower Bound	168	401	569	–	–
W (10 iter.)	267	455	722	–	12.75
WI (10 iter.)	267	452	719	–	11.14

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

Instance 10:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	482	799	1281	–	14.91
SSD	526	780	1306	−1.95	23.66
SPD	399	731	1130	11.79	25.11
IPD (4 iter.)	366	731	1097	14.36	53.35
Lower Bound	312	703	1015	–	–
W (10 iter.)	470	792	1262	–	12.75
WI (7 iter.)	470	788	1258	–	10.16

Instance 11:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	895	1134	2029	–	14.70
SSD	804	1144	1948	3.99	24.18
SPD	825	1034	1859	8.38	25.67
IPD (5 iter.)	721	944	1665	17.94	61.72
Lower Bound	682	920	1602	–	–
W (10 iter.)	757	1010	1767	–	12.75
WI (8 iter)	746	976	1722	–	9.98

Instance 12:

Approach	Aircraft Delay	Crew Delay	Total Delay	% Improvement on B	Time (s)
B	446	616	1062	–	15.04
SSD	437	604	1041	1.98	19.58
SPD	442	551	993	6.50	20.01
IPD (8 iter.)	380	544	924	12.99	72.97
Lower Bound	347	532	879	–	–
W (10 iter.)	440	574	1014	–	12.75
WI (10 iter.)	440	562	1002	–	11.54

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

Instance	$\frac{(SSD-SPD)}{SSD}$ $\times 100\%$	$\frac{(SPD-IPD)}{SPD}$ $\times 100\%$	$\frac{(SSD-IPD)}{SSD}$ $\times 100\%$	$\frac{(W-IPD)}{W}$ $\times 100\%$	$\frac{(WI-IPD)}{WI}$ $\times 100\%$
1	2.34	3.73	5.99	4.75	2.43
2	2.08	5.25	7.22	9.73	9.36
3	0.43	9.63	10.02	9.63	6.35
4	8.36	4.43	12.42	12.42	9.76
5	7.20	10.10	16.55	12.03	11.11
6	5.27	3.51	8.59	4.78	3.58
7	2.26	7.60	9.69	5.51	3.81
8	3.10	6.84	9.73	8.27	7.96
9	−2.51	8.50	6.20	12.05	11.68
10	13.48	2.92	16.00	13.07	12.80
11	4.57	10.44	14.53	5.77	3.31
12	4.61	6.95	11.24	8.88	7.78
Average	4.27	6.67	10.68	8.91	7.49

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 6.7% over the SPD approach.
- For the two sequential approaches tested, accurately calculating propagated delay is an improvement over using a simpler additive delay; 11 out of the 12 instances showed an improvement. On average over the 12 instances, the SPD approach improves over SSD by 4.3%.
- 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 10.7%.

When comparing our IPD approach with the methodology of [25] on average our approach produced schedules with 8.91% less total delay (IPD vs. W) and 7.49% less total delay (IPD vs. our "improved" version of [25] WI). The delay reductions over Algorithms W and WI are comparable to those observed by (i) the correct propagated delay was used in place of the simplified "summed" delay (SSD vs. SPD) and (ii) iteration was used in place of sequential optimisation (SPD vs. IPD).

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 3.9. 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 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. The authors gratefully acknowledge feedback provided by two anonymous referees that considerably improved the manuscript.

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. Shenoi. 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. Soloman, 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] Eurocontrol. Delays to Air Transport in Europe – Annual Report. 2006, 2009.

[12] IATA. Airport Handling Manual. 2010.

[13] 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.

[14] 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.

[15] A. Mercier, J. F. Cordeau, and F. Soumis. A Computational Study of Benders Decomposition for the Integrated Aircraft Routing and Crew Scheduling Problem. *Compututational Operations Research*, 32(6):1451–1476, 2005.

[16] A. Mercier and F. Soumis. An Integrated Aircraft Routing, Crew Scheduling and Flight Retiming Model. *Computers and Operations Research*, 34:2251–2265, 2007.

[17] N. Papadakos. Integrated Airline Scheduling. *Computers and Operations Research*, 36(1):176–195, 2009.

[18] Research and Innovative Technology Administration. Bureau of transportation statistics, 2010.

[19] B. Rexing, C. Barnhart, T. Kniker, A. Jarrah, and N. Krishnamurthy. Airline Fleet Assignment with Time Windows. *Transportation Science*, 34(1):1–20, 2000.

[20] 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.

[21] R. Sandhu and D. Klabjan. Integrated Airline Fleeting and Crew-Pairing Decisions. *Operations Research*, 55(3):439–456, 2007.

[22] A. Schaefer, E. L. Johnson, A. J. Kleywegt, and G. L. Nemhauser. Airline Crew Scheduling Under Uncertainty. *Transportation Science*, 39(3):340–348, 2005.

[23] B. Smith and E. L. Johnson. Robust Airline Fleet Assignment: Imposing station purity using station decomposition. *Transportation Science*, 40(4):497–516, 2006.

[24] 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.

[25] 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.

[26] C. L. Wu. Inherent Delays and Operational Reliability of Airline Schedules. *Air Transport Management*, 11:273–282, 2005.

[27] C. L. Wu. Improving Airline Robustness and Operational Reliability by Sequential Optimisation Algorithms. *Network and Spatial Economics*, 6:235–251, 2006.

[28] C. L. Wu. *Airline Operations and Delay Management - Insights from Airline Economics, Networks and Strategic Schedule Planning.* Ashgate, 2010. IN PRESS.

[29] J. W. Yen and J. R. Birge. A Stochastic Programming Approach to the Airline Crew Scheduling Problem. *Transportation Science*, 40(1):3–14, 2006.