

University of Wollongong

Research Online

SMART Infrastructure Facility - Papers

Faculty of Engineering and Information
Sciences

1-1-2014

An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing

Michelle E. Dunbar

University of Wollongong, mdunbar@uow.edu.au

Gary Froyland

University of New South Wales

Cheng-Lung Wu

University of New South Wales

Follow this and additional works at: <https://ro.uow.edu.au/smartpapers>



Part of the [Engineering Commons](#), and the [Physical Sciences and Mathematics Commons](#)

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au

An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing

Abstract

For reasons of tractability, the airline scheduling problem has traditionally been sequentially decomposed into various stages (e.g. 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 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. In Dunbar et al. (2012) [9] we introduced a new algorithm to accurately calculate and minimize the cost of propagated delay, in a framework that integrates aircraft routing and crew pairing. In this paper we extend the approach of Dunbar et al. (2012) [9] by proposing two new algorithms that achieve further improvements in delay propagation reduction via the incorporation of stochastic delay information. We additionally propose a heuristic, used in conjunction with these two approaches, capable of re-timing an incumbent aircraft and crew schedule to further minimize the cost of delay propagation. These algorithms provide promising results when applied to a real-world airline network and motivate our final integrated aircraft routing, crew pairing and re-timing approach which provides a substantially significant reduction in delay propagation.

Disciplines

Engineering | Physical Sciences and Mathematics

Publication Details

Dunbar, M., Froyland, G. & Wu, C. (2014). An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing. *Computers and Operations Research*, 45 68-86.

An Integrated Scenario-Based Approach for Robust Aircraft Routing, Crew Pairing and Re-timing

Michelle Dunbar*

SMART Infrastructure Facility, University of Wollongong, Sydney NSW 2522, Australia.

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. In Dunbar *et al.* [9] we introduced a new algorithm to accurately calculate and minimize the cost of propagated delay, in a framework that integrates aircraft routing and crew pairing. In this paper we extend the approach of [9] by proposing two new algorithms that achieve further improvements in delay propagation reduction via the incorporation of stochastic delay information. We additionally propose a heuristic, used in conjunction with these two approaches, capable of re-timing an incumbent aircraft and crew schedule to further minimize the cost of delay propagation. These algorithms provide promising results when applied to a real-world airline network and motivate our final integrated aircraft routing, crew pairing and re-timing approach which provides a substantially significant reduction in delay propagation.

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

1 Introduction

1.1 The airline scheduling problem

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 [17] states that approximately 21.5% of all flight legs in the U.S. between the months of July 2010 and July 2011 were delayed more than 15 minutes – with late arrivals and cancellations together accounting for approximately half of this delay.

In recent years, this has resulted in an ever-increasing discrepancy between planned costs and realized operational costs. As aircraft networks continue to grow, this trend is set to continue,

*A significant proportion of this research was completed while the author was with the University of New South Wales.

with AhmadBeygi *et al.* [1] 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.

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 reach 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.* (2010) [21]. The origin and destination of each flight leg (known as an O-D 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. The aircraft routing problem is often further decomposed into two sub-problems: aircraft routing generation and tail assignment. If performed sequentially by these two sub-problems, aircraft routing generation is usually modelled as a feasibility problem that produces generic maintenance feasible routings. As the day of operations draws closer, generic aircraft routings are then assigned for specific aircraft, i.e. tails, for operations. Finally the last stage, known as *crew pairing*, is also performed separately for each fleet type, as flight crew pilots are only certified for one fleet type at any given moment. 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 safety regulations (such as the 8-in-24 rule)¹, and ensure each flight is covered exactly once by exactly one crew group. Feasible pairings are then assigned to airline crew members as part of the crew rostering problem.

1.2 Integrated re-timing approaches

Recognizing that schedule generation plays a key role in determining the feasibility of subsequent aircraft and crew assignments, a number of authors have attempted to combine (an approximation of) schedule generation with fleet assignment, aircraft routing and crew pairing. Additionally, various authors have attempted to incorporate extra flexibility (and potentially improve operational robustness) via the introduction of *time windows*. Time windows allow the departure time to fall anywhere within a discretized window, usually extending 10 – 15 minutes either side of the originally scheduled departure time. This is achieved through the introduction of additional flight copies; each corresponding to a choice of possible departure times within the discretized time window, along with corresponding connection arcs; see Figure 1 below.

Desaulniers *et al.* (1997) [6] introduce time windows on flight departures for the fleet assignment problem. The problem is modelled as a multi-commodity flow in which extra time variables are introduced. The authors solve this problem using branch-and-bound and column generation, in which the column generator is a time-constrained shortest path problem.

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 [3] for further details.

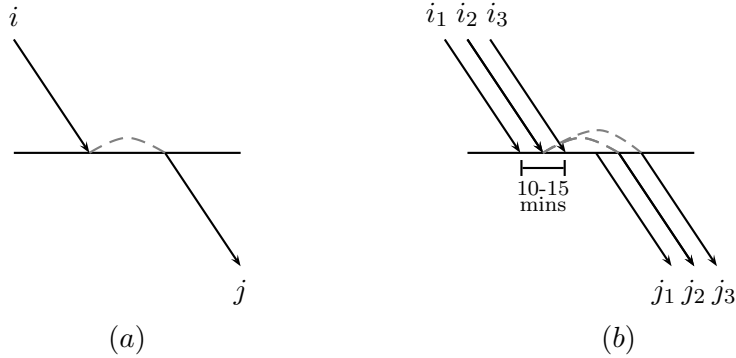


Figure 1: An illustration of flight copies used to allow changes to departure/arrival time within a 10-15 minute window. Flight i may now be represented by one of the flight copies i_1, i_2 and i_3 and similarly for flight j . Note that in (b) it is possible to connect each of the flight copies for i with each of the flight copies for j , provided that the slack is non-negative.

Rexing *et al.* (2000) [18] incorporate time windows within the fleet assignment problem, and the time windows are discretized into 5 (and 1) minute intervals. Klabjan *et al.* (2002) [11] address the problem of airline crew scheduling, incorporating time windows to allow more flexibility in the crew-pairing solution. Lan *et al.* (2006) [13] attempt to incorporate robustness into the schedule by estimating delays for each flight leg and minimizing expected delay. Lan *et al.* also use time windows (referred to as re-timing) to address the issue of reducing missed connections for passengers. As in the above, they introduce flight copies, and estimate the number of disrupted passengers for each possible connection using a connection-based model, leaving the fleet and routing solutions unchanged. Samardi *et al.* (2004) [19] extend the passenger connection model of Lan (2003) [12] and present an integrated flight departure, re-timing and aircraft routing model that aims to minimize the expected number of misconnecting passengers. Their model attempts to provide any potential misconnecting passengers with alternative recovery options. Lohatepanont *et al.* (2004) [14] extend the itinerary-based fleet assignment model (IFAM) of Barnhart *et al.* (2002) [4] to determine market service frequency, departure times and fleet assignments simultaneously. The authors make use of a set of flight legs that may be categorized as mandatory or optional, and assess the worth of a particular itinerary and re-adjust flight leg demand if a particular itinerary is removed, or if the schedule is altered. Belanger *et al.* (2006) [5] present an integrated model for fleet assignment with time windows for which they assume the schedule is periodic. The authors penalize short connections between flights and make use of profit estimations that integrate and capture both departure time and aircraft type; resulting in a potentially more profitable matching of fleet type with expected passenger demand. Klabjan *et al.* (2002) [11] partially integrate aircraft routing with crew scheduling. Solving the problem sequentially, the authors add plane-count constraints to the crew scheduling model to obtain a feasible aircraft routing problem. The authors also include time windows to allow more flexibility within the crew scheduling problem. More recently, Weide (2009) [20] proposed a model for the robust and integrated aircraft routing and crew pairing problem with time windows, employing two different methods for its solution. The model partially integrates scheduling decisions via the inclusion of departure time windows and seeks to achieve robustness by penalising aircraft changes for which connection time is less than a specified restricted time.

1.3 Outline of this paper

This paper is set out as follows. In Section 2 we introduce key notation and equations that are used extensively throughout this paper. In Section 3 we outline a re-timing heuristic that

improves upon the shortcomings of the existing re-timing approaches mentioned above via the *simultaneous* re-timing of aircraft and crew and in such a way as to minimize delay propagation between aircraft and crew. In Section 4 we extend the integrated aircraft routing and crew pairing model of Dunbar *et al.* [9] by proposing two alternative approaches for incorporating delay *scenarios* within the aircraft routing and crew pairing problems and outline an approach for integrating this with the improved heuristic. This inclusion of multiple delay scenarios allows an airline to incorporate historical primary delay information into the model in a more meaningful way, rather than simply making use of expected delays. Finally in Section 5 we improve upon these algorithms further by including scheduling decisions within the aircraft routing subproblem to provide even better quality solutions.

2 Preliminaries

In this paper we make use of a number of equations and algorithms proposed in Dunbar *et al.* (2012) [9]. For a complete overview of these algorithms and additional accompanying explanations, the reader is referred to [9]. We now introduce some key notation used throughout this paper. 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 two *primary delays* associated with it. 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 *expected* 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 sum of the *expected* en-route delay for flight i , and other crew-related expected primary delays during aircraft turnaround time, such as late crew boarding and crewing procedures. En-route 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 [10, 24]. 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. 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 (often referred to as a “string”) containing the connection (i, j) in the aircraft connection network:

$$d_j^R = \max \{ d_i^R - (s_{ij} - p_{ij}^R), 0 \}, \quad j \neq so, \quad (1)$$

and in the crew connection network:

$$d_j^P = \max \{d_i^P - (s_{ij} - p_{ij}^P), 0\}, \quad j \neq so. \quad (2)$$

For computational tractability we assume (as in Lan *et al.* (2006) [13]) that the primary delay is independent of the propagated delay. We now give an augmented definition of d_j^R and d_j^P that will be used throughout the rest of the paper. To calculate the propagated delay along an aircraft string, *taking into account propagated delays from crew* we inductively apply:

$$d_j^R = \max \{d_i^R - (s_{ij} - p_{ij}^R), d_k^P - (s_{kj} - p_{kj}^P), 0\}, \quad j \neq so, \quad (3)$$

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 . Similarly, to calculate the propagated delay along a crew string, *taking into account propagated delays from aircraft* we inductively apply:

$$d_j^P = \max \{d_i^P - (s_{ij} - p_{ij}^P), d_k^R - (s_{kj} - p_{kj}^R), 0\}, \quad j \neq so. \quad (4)$$

Using equations (3) and (4) we calculate the propagated delay along each aircraft (resp. crew) string so as to determine the delay propagation cost for each string. Algorithm (2.1) below consistently updates both aircraft and crew propagated delays to allow for the accurate calculation of delay propagation cost.

Algorithm 2.1: Propagated Delay Evaluation Algorithm

Input: An incumbent aircraft routing and crew pairing solution. The delay and slack information for each connection in the network

Output: The total propagated delay in the network.

begin

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 crewing solution, update d_j^R and d_j^P together by inductively applying equations (3) and (4) moving strictly forwards throughout the day.

end

2.1 Solving the Integrated Aircraft Routing and Crew Pairing Problem

As in Dunbar *et al.* (2012) [9], we state the standard aircraft routing problem as

$$\begin{aligned} \text{minimize:} \quad & (\mathbf{c}^R)^T \mathbf{x}^R \\ \text{Subject to:} \quad & A^R \mathbf{x}^R = \mathbf{e} \\ & \sum_{i=1}^{n_R} x_i^R \leq N \\ & \mathbf{x}^R \in \{0, 1\}^{n_R} \end{aligned} \quad (5)$$

where \mathbf{c}^R denotes a vector containing the aircraft delay propagation costs for each aircraft string. 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. The decision variable x_j^R takes the value 1 if routing j is included in the optimal solution and 0 otherwise. Finally, \mathbf{e} is an m -dimensional column vector of 1s and N is an upper bound on the number of aircraft available. The crew pairing problem may be stated in a similar manner, as the aircraft routing problem. The crew 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 the number of feasible crew pairings. The element c_j^P denotes the cost of column j and is defined as for the aircraft routing problem above. We additionally impose an upper limit H on the number of hours worked, an upper bound M on the number of available crew, and the restriction that crew must return to the base of origin.

$$\begin{aligned}
& \text{minimize:} && (\mathbf{c}^P)^T \mathbf{x}^P && (6) \\
& \text{Subject to:} && A^P \mathbf{x}^P = \mathbf{e} \\
& && \sum_{i=1}^{n_P} x_i^P \leq M \\
& && \mathbf{x}^P \in \{0, 1\}^{n_P}
\end{aligned}$$

We now outline our solution approach for the integrated aircraft routing and crew pairing problem and the calculation of the combined effects of delay propagation. It is assumed that there is no recovery of flights and that all flight departures will be pushed back accordingly, until the required resources are available.

Let $\pi(i)$ denote an ordered collection of nodes in the aircraft path π , truncated so that node i is the final node in the list. Additionally, we use the notation $\pi^-(i)$ to denote the node prior to node i in path π . We make use of the same notation for a crew path ξ . Finally, for the aircraft 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 delays at node j are given by

$$d_j^R = \max\{d_l^R - (s_{lj} - p_{lj}^R), d_k^P - (s_{kj} - p_{kj}^P), 0\}, \quad (7)$$

$$d_j^P = \max\{d_k^P - (s_{kj} - p_{kj}^P), d_l^R - (s_{lj} - p_{lj}^R), 0\}. \quad (8)$$

Suppose that in the current routing pricing problem we consider replacing the current aircraft connection (l, j) with (i, j) . We calculate d_j^R along the routing string being constructed using (3). If this potential replacement string is inserted into the master basis of (5), there will be an impact on the crew delays. Using (4), at node j , the new (locally calculated) crew delay is given by

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

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 consideration. For further details, the reader is referred to Section 2.5 of [9]. Denote by $d_{\pi(i)}^R$ the propagated expected routing delay at node i , computed along path $\pi(i)$ using (3), and by a_j^R and a_j^P , the costs per unit of delay at node j for the aircraft and crew respectively. For the purposes of this paper, we set each $a_j^R = 1$ and $a_j^P = 1$ and use the terms delay and delay cost, interchangeably. To simplify notation, we thus define the total delay at node i by $A_{\pi(i)}^R = \sum_{j \in \pi(i)} a_j^R d_{\pi(j)}^R$. 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 (5) and by $-w_i^P$ the weights from the pairing master (6). Define $W_{\pi(i)}^R = \sum_{j \in \pi(i)} (w_j^R + a_j^P(\tilde{d}_{j;\pi^-(j)}^P - d_{\xi(j)}^P))$.

The aircraft routing problem may be solved using column generation techniques and a label

setting algorithm outlined in Algorithm 2.2 which minimizes:

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

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

$$\begin{aligned} c_j^R &= z^R - \sum_{i \in \pi} w_i^R, \\ &= \sum_{i \in \pi} \left(a_i^R d_{\pi(i)}^R + a_i^P (\tilde{d}_{i;\pi^-(i)}^P - d_{\xi(i)}^P) \right), \end{aligned} \quad (11)$$

where the second equality uses (10). We now outline our label setting solution algorithm, augmented by a notion of label dominance, modified from related problems in Desrochers and Soumis (1988) [7] and Dumitrescu and Boland (2003) [8], that works efficiently in the cases tested. The aircraft routing problem makes use of the following dominance condition defined in [9]:

Definition 2.1. (*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.

Definition 2.2. (*Efficient Label*)

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 to which it corresponds at node i , is efficient.

Algorithm 2.2: Label Setting (Pricing) Algorithm for the Aircraft Routing Problem

1. *Initialisation:*
Set $I_{so} = \{so\}$ and $I_i = \emptyset$ for all $i \in \mathcal{N} \setminus \{so\}$.
Set $M_i = \emptyset$ for each $i \in \mathcal{N}$.
 2. *Selection of the label to be treated:*
if $\bigcup_{i \in \mathcal{N}} (I_i \setminus M_i) = \emptyset$ **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 $\arg \min_{\pi(t) \in I_t} A_{\pi(t)}^R + W_{\pi(t)}^R$.
-

In a similar manner, the crew pairing problem may also be solved using column generation techniques and a label setting algorithm similar to that of Algorithm 2.2. For the crew pairing

pricing problem, we additionally impose an upper limit H on the number of hours worked, and the restriction that the crew must return to the base of origin:

$$z^P = \min \left\{ \sum_{i \in \xi} \left(a_i^P d_{\xi(i)}^P + w_i^P + a_i^R (\tilde{d}_{i;\xi^-(i)}^R - d_{\pi(i)}^R) \right) : \begin{array}{l} \xi \text{ is a path from } so \text{ to } t, \\ \text{total hours worked} \leq H. \end{array} \right\}, \quad (12)$$

Upon obtaining a solution to (12), the minimizing path (or string) forms a column A_j of the matrix A^P . A crew pairing string is assigned a cost of

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

We now describe how to modify Algorithm 2.2 in order to solve the corresponding problem (12) for the crew. Define $T_{\xi(i)} = \sum_{j \in \xi(i)} t_j$, 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 (12) can be written as

$$z^P = \min \left\{ A_{\xi(t)}^P + W_{\xi(t)}^P : \xi \text{ is a path from } so \text{ to } t, T_{\xi(t)} \leq H \right\}. \quad (14)$$

In summary, the integrated aircraft routing and crew pairing algorithm of [9] is given below:

Algorithm 2.3: Iterative Aircraft Routing and Crew Pairing Algorithm (IPD)

1 INITIALISATION:

- (a). Solve problems (5) and (6) 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 $d_k^P = 0$, $d_k^R = 0$ 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 2.1.
- (b). Assign delay costs to strings using (11). Solve problem (5) 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 2.1.
- (b). Assign delay costs to strings using (13). Solve problem (6) 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, goto Step 5.

5 Return $\sum_{n=1}^N \sum_{i \in \pi_n^R} a_i^R d_{\pi(i)}^R + \sum_{m=1}^M \sum_{i \in \pi_m^P} a_i^P d_{\xi(i)}^P$, where π_n^R is the routing string for the n^{th} aircraft, $n = 1, \dots, N$ and π_m^P is the crew pairing string for the m^{th} crew, $m = 1, \dots, M$.

In the next section we outline a number of re-timing heuristics that are used in conjunction with the iterative aircraft routing and crew pairing algorithm of [9].

3 A heuristic re-timing algorithm (H)

The integrated aircraft routing and crew pairing model proposed in Dunbar *et al.* [9] finds an approximate minimal cost solution (with respect to delay propagation) for the integrated aircraft routing and crew pairing model, given a set of feasible connections, corresponding slack times for aircraft and crew, and corresponding primary delays for each of these connections. However, an optimal solution to this problem is the optimal assignment of aircraft and crew to flights, for the provided (*fixed*) set of departure times and list of feasible connections.

It was observed in [9] that the integrated algorithm has the potential to significantly improve (minimize) the total delay propagation cost over that of existing models. A question naturally arises as to whether one could improve the solution further, via the adjustment of the flight departure times, which would provide more slack over critical connections and draw excess slack from the remaining connections. Such an adjustment must reflect a real-world network and although it may be easy to minimize delay propagation by simply padding the schedule with slack over each connection; such a solution is expensive in practice, as resources remaining idle represent lost revenue for the airline. Furthermore, in order to ensure that the fleet assignment (completed prior to the aircraft routing and crew pairing) remains feasible, we wish to keep such a re-timed schedule close to the original schedule.

In this section we address whether it is possible to improve the solution further, via the adjustment of the flight departure times and propose a greedy heuristic capable of re-timing an incumbent aircraft and crew assignment in order to minimize the cost of delay propagation. The heuristic re-times flights without altering the aircraft and crew assignments of the incumbent solution. The algorithm is greedy in the sense that it moves forward in time throughout the day, starting with the very first flight in each route (or pairing) and then moves on to the second and so on. Because the algorithm proceeds forward in time, changes made later in the day do not affect earlier changes made. To guide the algorithm toward an optimal solution, we specify that the algorithm accepts only improvements in the overall delay and if no improvement can be made, we do not re-time the flight under consideration.

3.1 Assumptions

In order for the re-timing heuristic to obtain a solution that reflects real-world operations, we are required to make several key assumptions; specifically, these restrictions are placed on the amount by which we can alter the departure times, and therefore the corresponding slack, over certain connections and in the network as whole.

Firstly, as noted above, monetary factors relating to aircraft and crew costs (eg. costs of obtaining new slot times or overtime costs for crew) make excess ground time expensive and undesirable as idle resources represent lost revenue for the airline. Consequently, the associated costs of lost revenue may outweigh the benefits of the potential for delay absorption. Thus, in our algorithm we ensure that we only make use of the current slack present in the network. Therefore the algorithm must only draw from the slack already present in the network and attempt to place it where it is required the most. We seek to avoid significant changes to the publicized (incumbent) timetable so as to (i) ensure both the aircraft routing and crew pairing solutions remain feasible and (ii) the demand for each flight remains approximately the same, so that the fleet assignment remains optimal. Therefore, we use the idea of restricting the amount by which we can re-time each flight to within a time-window; an idea successfully employed by a number of authors including, Rexing *et al.* (2000) [18], Lan *et al.* (2006) [13] and Mercier and Soumis (2007) [15]. For simplicity, we assume a time window of $[t - 10, t + 10]$ around the originally scheduled departure time t , discretized into 5 minute intervals.

We assume that changes made to slack over a connection are the same for aircraft and crew. We conclude by mentioning that by moving a flight earlier we are effectively reducing slack, and

by moving a flight later we increase the slack for that flight. Thus if we add slack to a certain flight connection, we subtract the same amount of slack from the following flight connection in the route (or pairing) as the first flight will arrive later - thus our algorithm involves only a localized effect. Furthermore, slack is only taken from within the same aircraft route (resp. crew pairing). Both schedules are re-timed via the iteration between one feasible aircraft schedule to another feasible crew schedule. The feasibility of the proposed change is ensured by the if clause present in both the aircraft routing and crew pairing sections of Algorithm 3.1.

In this heuristic we evaluate our choice of departure time over $|\Omega|$ delay scenarios, $\omega \in \Omega$ for each connection, where Ω denotes the set of all delay scenarios. We introduce a primary delay for the aircraft across connection (i, j) under scenario $\omega \in \Omega$ by $p_{ij,\omega}^R$. We use analogous notation $p_{ij,\omega}^P$, to denote primary crew delays for each connection. As flights from the source so , and sink t , are ‘dummy’ flights, we set $p_{jt,\omega}^R = 0$ and $p_{jt,\omega}^P = 0$ for all $(j, t) \in \mathcal{A}$.

Algorithm 3.1: The re-timing heuristic (H) using $|\Omega|$ different delay scenarios

Input: An incumbent aircraft routing and crew pairing solution, and a set of primary delays $\{p_{ij,\omega}^R\}_{ij \in \mathcal{A}}$, $\{p_{ij,\omega}^P\}_{ij \in \mathcal{A}}$ for all $\omega \in \Omega$ with corresponding slack $\{s_{ij}^R\} = \{s_{ij}^R\}_{ij \in \mathcal{A}}$ and $\{s_{ij}^P\} = \{s_{ij}^P\}_{ij \in \mathcal{A}}$, over each connection for the aircraft and crew.

Output: An improved choice of slack for each feasible connection in the network, corresponding to an adjustment in departure time for each flight, leading to a reduced average delay cost.

```

1 Start AR
2 Set  $l := 1$  (the first flight in each string) and set slackOptions =  $[-10, -5, 0, 5, 10]$ .
3 for  $i$  from 1 to numberOfAircraft do
4   Pick flight string  $i$ .
5   for  $j$  from 1 to numSlackOptions do
6     - Set  $s_{opt} := \text{slackOptions}[j]$ 
7     - Find the flight  $k$  that precedes flight  $l$  in string  $i$ . (N.B:  $k = 0$ , if  $l = 1$ )
8     - Find the flight  $m$  that follows flight  $l$  in string  $i$ .
9     - Construct the temporary slack vectors  $\hat{s}^R := s^R$  and  $\hat{s}^P := s^P$ .
10    if  $(\hat{s}_{k,l}^R + s_{opt}) \geq 0$  and  $(\hat{s}_{k',l}^P + s_{opt}) \geq 0$  and  $(\hat{s}_{l,m}^R - s_{opt}) \geq 0$  and  $(\hat{s}_{l,m'}^P - s_{opt}) \geq 0$  then
11      - Set  $\hat{s}_{k,l}^R := \hat{s}_{k,l}^R + s_{opt}$ . (resp.  $\hat{s}_{k',l}^P := \hat{s}_{k',l}^P + s_{opt}$ )
12      - Set  $\hat{s}_{l,m}^R := \hat{s}_{l,m}^R - s_{opt}$ . (resp.  $\hat{s}_{l,m'}^P := \hat{s}_{l,m'}^P - s_{opt}$ )
13      forall the  $\omega \in \Omega$  do
14        | Apply Algorithm (2.1) using  $\hat{s}^R$  and  $\hat{s}^P$  and store the corresponding total delay cost.
15      end
16      - Calculate the average delay over all the simulations and record this average delay cost with
17        its corresponding slack option.
18    end
19  end
20  - Choose the best (i.e. smallest delay cost) slack option ( $s_{opt}^*$ ) and update the real slack vector:
21  - Set  $s_{k,l}^R := s_{k,l}^R + s_{opt}^*$ . (resp.  $s_{k',l}^P := s_{k',l}^P + s_{opt}^*$ )
22  - Set  $s_{l,m}^R := s_{l,m}^R - s_{opt}^*$ . (resp.  $s_{l,m'}^P := s_{l,m'}^P - s_{opt}^*$ )
23  - Set  $i := i + 1$ .
24 end
25 StartCP
26 for  $i$  from 1 to numberOfCrew do
27   Pick flight string  $i$ .
28   for  $j$  from 1 to numSlackOptions do
29     - Set  $s_{opt} := \text{slackOptions}[j]$ .
30     - Find the flight  $k$  that precedes flight  $l$  in string  $i$ .
31     - Find the flight  $m$  that follows flight  $l$  in string  $i$ .
32     - Construct the temporary slack vectors  $\hat{s}^R := s^R$  and  $\hat{s}^P := s^P$ .
33     if  $(\hat{s}_{k,l}^R + s_{opt}) \geq 0$  and  $(\hat{s}_{k',l}^P + s_{opt}) \geq 0$  and  $(\hat{s}_{l,m}^R - s_{opt}) \geq 0$  and  $(\hat{s}_{l,m'}^P - s_{opt}) \geq 0$  then
34       - Set  $\hat{s}_{k',l}^P := \hat{s}_{k',l}^P + s_{opt}$ . (resp.  $\hat{s}_{k,l}^R := \hat{s}_{k,l}^R + s_{opt}$ )
35       - Set  $\hat{s}_{l,m'}^P := \hat{s}_{l,m'}^P - s_{opt}$ . (resp.  $\hat{s}_{l,m}^R := \hat{s}_{l,m}^R - s_{opt}$ )
36       forall the  $\omega \in \Omega$  do
37         | Apply Algorithm (2.1) using  $\hat{s}^R$  and  $\hat{s}^P$  and store the corresponding total delay cost.
38       end
39       - Calculate the average delay cost over all the simulations and record this average delay cost
40         with its corresponding slack option.
41     end
42   end
43   - Choose the best (i.e. smallest delay cost) slack option ( $s_{opt}^*$ ) and update the real slack vector:
44   - Set  $s_{k,l}^P := s_{k,l}^P + s_{opt}^*$ . (resp.  $s_{k',l}^R := s_{k',l}^R + s_{opt}^*$ )
45   - Set  $s_{l,m}^P := s_{l,m}^P - s_{opt}^*$ . (resp.  $s_{l,m'}^R := s_{l,m'}^R - s_{opt}^*$ )
46   - Set  $i := i + 1$ .
47 end
48 - Set  $l := l + 1$ . (Move on to the next flight for each string)
49 Return to 1.

```

3.2 Description of key contributions

Our proposed heuristic is used in conjunction with Algorithm 2.3 and seeks to improve upon existing re-timing approaches in a number of key areas. Firstly, many existing methods that incorporate an element of re-timing do so for either the aircraft or crew in isolation. For example, in the model of Klabjan *et al.* (2002) [11] and Lan *et al.* (2006) [13], the authors allow re-timing possibilities for the aircraft routing solution (resp. crew pairing solution), but do not consider crew pairing (resp. aircraft routing). More recently the models proposed by Papadakos (2007) [16] and Mercier and Soumis (2007) [15] include re-timing possibilities for both aircraft and crew; however neither of these approaches consider the effects of delay propagation between aircraft and crew, and involve the addition of a large number of extra decision variables within the master problem corresponding to a range of alternative departure times which may potentially lead to lengthy computation times.

Our proposed heuristic improves upon these approaches by *simultaneously* re-timing the aircraft and crew whilst preserving the aircraft and crew assignments of an incumbent solution. The simultaneous re-timing of aircraft and crew avoids complications arising from potential incompatibilities and allows for *accurate delay propagation calculation* between aircraft and crew. This provides an accurate assessment of the *effects of delay propagation* across certain connections and thus allows an effective re-timing to be chosen. This approach additionally overcomes the difficulties associated with the heuristic of AhmadBeygi (2010) [1] which both under-estimates and over-estimates the effects of delay propagation in certain cases.

The primary advantage of preserving the assignments of the incumbent solution is that it ensures the re-timed solution for aircraft and crew does not differ significantly from the original aircraft and crew assignment. This may be beneficial for an airline that has aircraft routes or pairings that don't change substantially from day to day. Preserving assignments of aircraft and crew also allows the heuristic to either be used in conjunction with a model such as the one proposed in Dunbar *et al.* (2011) [9], or as an 'add-on' to an existing model, allowing for a qualitative assessment of potential areas for improvement within a given solution (e.g. bottlenecks at certain points) and providing schedule planners an insight into where improvements can be made. The algorithm can be easily incorporated into an iterative process between the aircraft routing, crew pairing and re-timing *without increasing the overall complexity*. This possibility is discussed in further detail in the following sections. Finally, an airline may easily specify flights that it would prefer not to be re-scheduled (eg. first flights of the day).

4 Incorporating scenarios within the subproblems

In the previous section we motivated the concept of re-timing within the context of airline scheduling and proposed a re-timing heuristic that may be used in conjunction with the IPD approach of Algorithm 2.3 above. In Section 6 we will demonstrate that this combination delivers significant improvements in delay propagation reduction. However, one of the drawbacks of the IPD approach (Algorithm 2.3) is the assumption that the expected delay over each connection is known prior to the assignment of aircraft and crew to flights, with delay propagation calculations utilizing the expected delay over each connection. Furthermore, the IPD approach can *only* make use of expected delays and no other additional information.

In practice, airlines may possibly have this data at their disposal, and require the development of new mathematical tools and techniques in order to incorporate this information in a meaningful way. Alternatively, given the number of sources of primary delay over a particular connection, an airline may prefer to model delay across individual connections using a distribution (as mentioned in Wu (2005, 2007) [23, 22]) – perhaps derived from data from previous years, to represent a more complete range of delays observed.

With these issues in mind, we are thus motivated to investigate whether it is possible to achieve further improvements in delay propagation reduction by incorporating stochastic delay information within the aircraft and crew label setting algorithms. We propose to incorporate this information by prescribing $|\Omega|$ potential primary delay values for each connection in the network. These $|\Omega|$ primary delay values might be drawn from an appropriate distribution (i.e. truncated normal or exponential distribution) possessing the same mean as the original primary delay prescribed in the test instances of [9]. We now outline two approaches for the inclusion of multiple delay scenarios within the subproblem.

The first approach, referred to as Exact, is an exact approach in which we enumerate every feasible aircraft and crew path from source to sink in the aircraft and crew subproblems and then for each path, calculate the average delay propagation along each path, over all delay scenarios. In Section 6 we show that this approach produces very good results on our network, but may become computationally expensive for larger networks. To reduce the computational cost, we propose a second approach, referred to as Local, in which at each step of the label setting algorithm, we calculate the average delay propagation arriving at each node and then use this to decide which label(s) to propagate further. This has the advantage of potentially fewer labels and paths being produced and we are not required to enumerate all possible paths. Both approaches perform well and outperform the IPD case in which we simply use mean delays. We now formalize the above description and detail each approach.

4.1 Notation

Define $d_{\pi(i),\omega}^R$ to be the propagated delay at node i under scenario ω for path π . We introduce the notation $a_i^P(\tilde{d}_{i,\omega;\pi^-(i)}^P - d_{\xi(i),\omega}^P)$ to denote the *estimate* of the reduced cost for crew delay attributable to node i , under scenario ω with $\omega \in \Omega$.

4.2 The Exact (E) approach: pricing problem

In this first approach, we enumerate every feasible path between the source node so , and sink node t , and determine the average delay propagation along each path over all scenarios. The algorithm then finds the path that minimizes:

$$z^R = \min \left\{ \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \pi} \left(a_i^R d_{\pi(i),\omega}^R + w_i^R + a_i^P (\tilde{d}_{i,\omega;\pi^-(i)}^P - d_{\xi(i),\omega}^P) \right) : \pi \text{ is a path from } so \text{ to } t \right\}, \quad (15)$$

For the crew pricing problem, we analogously use the reduced cost estimate $a_j^R(\tilde{d}_{j,\omega;\pi^-(j)}^R - d_{\pi(j),\omega}^R)$ 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, with the further restriction that the path ξ begins and ends at the same crew base.

$$z^P = \min \left\{ \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \xi} \left(a_i^P d_{\xi(i),\omega}^P + w_i^P + a_i^R (\tilde{d}_{i,\omega;\xi^-(i)}^R - d_{\pi(i),\omega}^R) \right) : \begin{array}{l} \xi \text{ is a path from } so \text{ to } t, \\ \text{total hours worked} \leq H. \end{array} \right\}, \quad (16)$$

Upon obtaining a solution to the pricing problem (15) (resp. problem (16)), the minimizing path forms a column A_j of the matrix A^R (resp. A^P). A routing string is assigned a cost of

$$c_j^R = z^R - \sum_{i \in \pi} w_i^R = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \pi} \left(a_i^R d_{\pi(i),\omega}^R + a_i^P (\tilde{d}_{i,\omega;\pi^-(i)}^P - d_{\xi(i),\omega}^P) \right). \quad (17)$$

and a crew pairing string is assigned a cost of

$$c_j^P = z^P - \sum_{i \in \xi} w_i^P = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \xi} \left(a_i^P d_{\xi(i),\omega}^P + a_i^R (\tilde{d}_{i,\omega;\xi^-(i)}^R - d_{\pi(i),\omega}^R) \right). \quad (18)$$

We now outline the algorithms for the aircraft routing and crew pairing subproblems used in the Exact case:

Algorithm 4.1: Algorithm for the Aircraft Routing Pricing Problem: Exact

1 *Generate all paths:*

Consider the full directed graph $G = (\mathcal{N}, \mathcal{A})$. Generate all *distinct, directed* paths from the source node so , to the sink node t .

2 Record each path π in the set S of all paths.

3 For each path $\pi \in S$, calculate

$$\frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \pi} \left(a_i^R d_{\pi(i),\omega}^R + w_i^R + a_i^P (\tilde{d}_{i,\omega;\pi^-(i)}^P - d_{\xi(i),\omega}^P) \right),$$

and return the path that minimizes (over all paths $\pi \in S$):

$$z^R = \min \left\{ \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \pi} \left(a_i^R d_{\pi(i),\omega}^R + w_i^R + a_i^P (\tilde{d}_{i,\omega;\pi^-(i)}^P - d_{\xi(i),\omega}^P) \right) \right\},$$

where π is a path from so to t .

Algorithm 4.2: Algorithm for the Crew Pairing Pricing Problem: Exact

1 *Generate all paths:*

Consider the full directed graph $G = (\mathcal{N}, \mathcal{A})$. Generate all *distinct, directed* paths from the source node so , to the sink node t , that satisfy:

- The total number of hours worked $\leq H$ **and**,
- The last flight in the path returns to the crew-base at which the path began.

2 Record each path ξ in the set S of all paths.

3 For each path $\xi \in S$, calculate:

$$\frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \xi} \left(a_i^P d_{\xi(i),\omega}^P + w_i^P + a_i^R (\tilde{d}_{i,\omega;\xi^-(i)}^R - d_{\pi(i),\omega}^R) \right),$$

and return the path that minimizes (over all paths $\xi \in S$):

$$z^P = \min \left\{ \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \sum_{i \in \xi} \left(a_i^P d_{\xi(i),\omega}^P + w_i^P + a_i^R (\tilde{d}_{i,\omega;\xi^-(i)}^R - d_{\pi(i),\omega}^R) \right) \right\},$$

where ξ is a path from so to t and total hours worked $\leq H$.

4.3 The Local (L) approach

In this second approach, we make use of the label setting algorithm (Algorithm 2.2). In contrast with Exact, we wish to incorporate delay information from the scenarios *within* the label setting

algorithm used in the pricing problem, rather than only use delay information once the paths have been calculated. This allows the average delay propagation cost at each node to be calculated before the label is propagated further. Furthermore, through use of the dominance condition (Definition 2.1), we encourage the propagation of labels whose corresponding (partial) path experiences approximately minimal propagated delay.

In this approach we once again make use of the scenarios consisting of $|\Omega|$ primary delays, for each connection. We assume that each scenario is equally likely, but this may be modified accordingly, if necessary. In this modification of the label setting aircraft routing and crew pairing algorithms, we wish to retain Steps 1 and 2 of Algorithm 2.2, but modify Step 3 to allow for the average delay propagation arriving at a particular node to be locally calculated. We thus use the notation \hat{d}_j to denote the average delay arriving at node j and more specifically, $\hat{d}_{\pi(j)}^R$ for the average propagated aircraft delay along path π and $\hat{d}_{\xi(j)}^P$ for the average propagated crew delay along path ξ . Since the source node so , is a ‘dummy’ node, we fix $\hat{d}_{so}^R = 0$ and $\hat{d}_{so}^P = 0$ and for a general flight node $j \neq 0$, we calculate the average propagated delay arriving at node j , along path π denoted by $\hat{d}_{\pi(j)}^R$ as follows:

$$\begin{aligned} \hat{d}_{\pi(j)}^R &= \mathbb{E}_\omega \left(\max\{\hat{d}_{\pi(i)}^R + (s_{ij} - p_{ij,\omega}^R), \hat{d}_{\xi(k)}^P + (s_{kj} - p_{kj,\omega}^P)\} \right), \\ &= \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \left(\max\{\hat{d}_{\pi(i)}^R + (s_{ij} - p_{ij,\omega}^R), \hat{d}_{\xi(k)}^P + (s_{kj} - p_{kj,\omega}^P)\} \right). \end{aligned} \quad (19)$$

Similarly, we calculate the average propagated crew delay arriving at node j , along path ξ denoted by $\hat{d}_{\xi(j)}^P$ as follows.

$$\begin{aligned} \hat{d}_{\xi(j)}^P &= \mathbb{E}_\omega \left(\max\{\hat{d}_{\pi(i)}^R + (s_{ij} - p_{ij,\omega}^R), \hat{d}_{\xi(k)}^P + (s_{kj} - p_{kj,\omega}^P)\} \right), \\ &= \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \left(\max\{\hat{d}_{\pi(i)}^R + (s_{ij} - p_{ij,\omega}^R), \hat{d}_{\xi(k)}^P + (s_{kj} - p_{kj,\omega}^P)\} \right). \end{aligned} \quad (20)$$

The modified aircraft and crew label setting algorithms for the Local approach are outlined below.

Algorithm 4.3: Label Setting Algorithm for the Aircraft Routing Problem: Local

1. *Initialisation:*
Set $I_{so} = \{so\}$ and $I_i = \emptyset$ for all $i \in \mathcal{N} \setminus \{so\}$.
Set $M_i = \emptyset$ for each $i \in \mathcal{N}$.
 2. *Selection of the label to be treated:*
if $\bigcup_{i \in \mathcal{N}} (I_i \setminus M_i) = \emptyset$ **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 $\hat{A}_{\pi(i)}^R + W_{\pi(i)}^R$ is minimal.
 3. *Treatment of label* $(\hat{A}_{\pi(i)}^R + W_{\pi(i)}^R, \hat{d}_{\pi(i)}^R)$
forall $(i, j) \in \mathcal{A}$
Calculate the \hat{d}_j^R and \hat{d}_j^P using equations (19) and (20) over all scenarios.
if $(\hat{A}_{\{\pi(i),j\}}^R + W_{\{\pi(i),j\}}^R, \hat{d}_{\{\pi(i),j\}}^R)$ is not dominated by $(\hat{A}_{\eta(j)}^R + W_{\eta(j)}^R, \hat{d}_{\eta(j)}^R)$ 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 $\arg \min_{\pi(t) \in I_t} \hat{A}_{\pi(t)}^R + W_{\pi(t)}^R$.
-

Algorithm 4.4: Label Setting Algorithm for the Crew Pairing Problem: Local

As in Algorithm 4.3, replacing R superscripts by P superscripts throughout and replacing the **if** clause in Step 3 with:

if $T_{\{\xi(i),j\}} \leq H$ hours **and** $(\widehat{A}_{\{\xi(i),j\}}^P + W_{\{\xi(i),j\}}^P, \widehat{d}_{\{\xi(i),j\}}^P, T_{\{\xi(i),j\}})$ is not dominated by $(\widehat{A}_{\eta(j)}^P + W_{\eta(j)}^P, \widehat{d}_{\eta(j)}^P, T_{\eta(j)})$ for any $\eta(j) \in I_j$
then set $I_j = I_j \cup \{\xi(i), j\}$

We now outline how these modifications to the aircraft routing and crew pairing subproblems may be included to form the **Exact** and **Local** approaches. The following Algorithms specify the necessary changes to be made to the IPD Algorithm (Algorithm 2.3).

Algorithm 4.5: Integrated AR and CP using **Exact** in each subproblem

- 1 Solve the integrated aircraft routing and crew pairing problem using Algorithm 2.3, making the following changes:
 - 2(b) Solve the aircraft routing problem (5) using column generation and assign delay costs using the **Exact** aircraft routing Algorithm 4.1.
 - 3(b) Solve the crew pairing problem (6) using column generation and assign delay costs using the **Exact** crew pairing Algorithm 4.2.
-

Algorithm 4.6: Integrated AR and CP using **Local** in each subproblem

- 1 Solve the integrated aircraft routing and crew pairing problem using Algorithm 2.3, making the following changes:
 - 2(b) Solve the aircraft routing problem (5) via column generation, using the **Local** Label Setting Algorithm 4.3.
 - 3(b) Solve the crew pairing problem (6) via column generation, using the **Local** Label Setting Algorithm 4.4.
-

After making the above changes to the subproblems, we now extend Algorithms 2.3, 4.5 and 4.6 above to include the re-timing heuristic (H), Algorithm 3.1. We propose three algorithms: IPD combined with H, **Exact** combined with H and **Local** combined with H.

Algorithm 4.7: Integrated AR, CP and re-timing: (IPD + H).

- 1 Solve the integrated aircraft routing and crew pairing problem using Algorithm 2.3.
 - 2 Apply the re-timing heuristic (H), Algorithm 3.1 to the incumbent AR and CP solutions.
 - 3 Apply the Propagated delay Algorithm 2.1 to the new solution to obtain the total delay.
-

Algorithm 4.8: Integrated AR, CP and re-timing: (**Exact** + H).

- 1 Solve the integrated aircraft routing and crew pairing problem using the **Exact** approach in each subproblem; that is, apply Algorithm 4.5.
 - 2 Apply the re-timing heuristic (H), Algorithm 3.1 to the incumbent AR and CP solutions.
 - 3 Apply the Propagated delay Algorithm 2.1 to the new solution to obtain the total delay.
-

Algorithm 4.9: Integrated AR, CP and re-timing: (Local + H).

- 1 Solve the integrated aircraft routing and crew pairing problem using the Local approach in each subproblem; that is, apply Algorithm 4.6.
 - 2 Apply the re-timing heuristic (H), Algorithm 3.1 to the incumbent AR and CP solutions.
 - 3 Apply the Propagated delay Algorithm 2.1 to the new solution to obtain the total delay.
-

In Section 6 we apply Algorithms 4.7, 4.8 and 4.9 to the current aircraft and crew pairing network using test delay instances sampled from the same distributions used in [9]. The results in Section 6 indicate that these algorithms, using modified subproblems in combination with the proposed heuristic, provided an improvement in solution quality and a reduction in the variance of the total delay over different delay scenarios.

4.4 Rescheduling the solution from each Approach + H

A natural extension of this algorithmic approach is to apply IPD, Exact and Local to the solutions obtained from the IPD + H (and resp. Exact + H and Local + H) respectively. In this subsection we consider the possibility of performing such an iteration and propose three new algorithms to investigate whether the solution quality may be improved even further. We outline each of these algorithms below and results for these algorithms are provided in Section 6.

Algorithm 4.10: Integrated AR, CP and re-timing: (IPD + H + IPD).

- 1 Apply Algorithm 4.7.
 - 2 Re-solve the integrated aircraft routing and crew pairing problem as outlined in Algorithm 2.3; using the new choice of departure times for each flight.
 - 3 Apply the Propagated delay Algorithm 2.1 to the new solution to obtain the total delay.
-

Algorithm 4.11: Integrated AR, CP and re-timing: (Exact + H + Exact).

- 1 Apply Algorithm 4.8.
 - 2 Re-solve the integrated aircraft routing and crew pairing problem; using the new choice of departure times for each flight using Algorithm 4.5.
 - 3 Apply the Propagated delay Algorithm 2.1 to the new solution to obtain the total delay.
-

Algorithm 4.12: Integrated AR, CP and re-timing: (Local + H + Local).

- 1 Apply Algorithm 4.9.
 - 2 Re-solve the integrated aircraft routing and crew pairing problem; using the new choice of departure times for each flight, using Algorithm 4.6.
 - 3 Apply the Propagated delay Algorithm 2.1 to the new solution to obtain the total delay.
-

5 Aircraft Routing, Crew Pairing and Re-timing (ARCPR)

5.1 Introduction and Motivation

In the previous sections, we outlined the Exact+H+Exact and Local+H+Local approaches for including scheduling decisions (alternative options for departure time) in the planning process. In Section 6 it may be observed that these algorithms deliver a marked improvement over simply using IPD, Exact or Local in isolation. Their primary computational advantage is that these algorithms do not require the addition of an extremely large number of decision variables to capture re-timing decisions and so avoid the computational complexity often associated with other problems. Algorithms 4.10, 4.11 and 4.12 scale as a standard aircraft routing or crew pairing problem. One of the drawbacks however, is that these algorithms only re-time a fixed aircraft and crew assignment, and so it may be possible to obtain better quality solutions by

incorporating this scheduling process directly within the integrated aircraft routing and crew pairing problem. That is, include all three decisions – aircraft routing, crew pairing and re-timing – within the one problem and solve them together, without increasing the complexity unnecessarily. In this section, we outline a new approach capable of embedding the scheduling decisions (choice of departure times for each flight) within the iterative, integrated framework of [9] whilst retaining the simple form of the aircraft routing and crew pairing formulations. We do this via an expanded network consisting of flight copies that represent different flight departure time choices. This expanded network exists only within the aircraft routing subproblem, and so does not increase the dimension (or complexity) of either the aircraft routing or crew pairing master problems (5) and (6). We now outline a few key assumptions.

5.2 Assumptions

Firstly, to ensure that our model reflects real world restrictions and that the original fleet assignment remains feasible, we only attempt to re-time flights to within a 10 minute window $[t - 10, t + 10]$ either side of the scheduled departure time, t . We create for each flight i , five duplicate flight nodes i_1, i_2, i_3, i_4, i_5 that lie within this discretized minute window. In Section 3 we tested our re-timing heuristic on a network consisting of 54 flights and so in this setup we will now have $54 \times 5 = 270$ (+2 corresponding to the source and sink) flight nodes in our subproblem.

This process of duplicating flight nodes is performed only within the subproblem, so as to ensure that we do not increase the complexity of the integrated iterative aircraft routing and crew pairing master problems (5) and (6). The process of duplication is done in a pre-processing step, and does not need to be repeated with each call to the subproblem. Moreover, we only perform this duplication within the *aircraft* routing subproblem, and assume that the schedule (departure times) chosen is also followed by the crew; thus eliminating potential conflicts between the aircraft and crew schedules. To ensure a fair comparison with the algorithms above, we do not wish to introduce any “new” arcs into the network. Rather, we only allow connections between flight nodes that correspond to connections in the original network. More specifically, if connection (i, j) was a feasible connection in the original network, we must allow all possible connection pairs (i_m, j_n) for $m = 1, \dots, 5, n = 1, \dots, 5$ in the new network (provided the slack across the connection is non-negative). The duplicated flight nodes are denoted by the set \mathcal{N}^* and the set of all feasible connections in this expanded network by \mathcal{A}^* , forming the augmented network graph $G^* = (\mathcal{N}^*, \mathcal{A}^*)$.

For each of these new connections (i_m, j_n) , we assume the primary delay p_{ij}^R , is the same as for the original connection (i, j) . Similarly, we assume that each flight copy i_1, i_2, \dots, i_5 in the subproblem inherits the dual value w_i as for flight i in the original network. Once the above pre-processing algorithm has been performed, along with the modifications to the primary delays and dual values, the aircraft routing subproblem is solved over this larger network using the label setting algorithm discussed in the previous sections. The solution (an aircraft string) to the aircraft routing subproblem is of length $54 \times 5 = 270$, which is converted back into the appropriate length of 54 for the master problem.

As our aircraft routing subproblem is now significantly larger than the original and we wish to solve the problem as quickly as possible, we make use of the **Local** approach in both the aircraft routing and crew pairing subproblems to avoid generating every possible path (as would occur in the **Exact** approach). Furthermore, it may be observed in Section 6 that the percentage improvement obtained by the **Local** approach was not significantly smaller than that of the **Exact** approach.

Algorithm 5.1: Integrated AR and CP using Local in each subproblem

- 1 Solve the integrated aircraft routing and crew pairing problem using Algorithm 2.3, making the following changes:
 - (i) Solve the aircraft routing problem (5) on the *expanded flight network* G^* via column generation, using the Local Label Setting Algorithm 4.3.
 - (ii) Convert the solution to the aircraft routing subproblem into a form compatible with the master problem (5) and update the slack information and departure times according to the choices made in the aircraft routing subproblem.
 - (iii) Solve the crew pairing problem (6) on the *original flight network* via column generation, using the Local Label Setting Algorithm 4.4.
-

6 Numerical Results

To evaluate the effectiveness of our proposed approaches, we apply each algorithm to a one-day schedule on a real airline network consisting of 54 flights and 128 feasible connections.

For simplicity we assume that all aircraft, crew and connections incur similar operating costs, and thus a “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 which will be the incumbent solution input to our iterative integrated approaches to reduce total propagated delay. We use 10 aircraft and 16 crew pairs in all instances and all algorithms tested (Algorithm 3.1 and those in Sections 4 and 5).

We test our new computational approaches on 12 random delay *instances*. Each instance consists of 20 *batches*, and each batch consists of 1000 delay *scenarios*. Each scenario consists of an independent sampling of delays for each connection in the network. Our 12 delay instances use scenarios drawn from an exponential distribution $E(\lambda)$ and a truncated normal distribution $tN(\mu, \sigma^2)$. More specifically, 3 instances use scenarios drawn 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 (in minutes) for all connections. We calculate the performance of our algorithm over the 20 batches in each instance in order to obtain an estimate of the statistics of the results for each instance.

We denote by the syntax **Approach + H** the results from the re-timing algorithm as applied to the incumbent solution using a specific **Approach**. We tabulate the results for each of the approaches discussed thus far. Each table provides a comparison with the Base Case and records results for **Approach + H**, **Approach + H + Approach** and the ARCPR approach. In each table we provide the time taken to solve each approach (in seconds) and specify the breakdown of the computational time required for each component explicitly; according to the order of implementation. In the tables below, we record the average delay and computational time required over the 20 different sets of 1000 delay scenarios, for each instance.

6.1 Exponential Distribution with $\lambda = 5$.

6.1.1 Instance 1

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	688	504	1192	–	9.17
IPD.	576	442	1018	14.60	47.19
Exact.	559	402	961	19.38	61.00
Local.	576	422	998	16.28	58.98
Base + H.	536	472	1008	15.44	9.17 + 21.16
IPD + H.	428	377	805	32.47	47.19 + 24.22
Exact + H.	415	343	758	36.41	61.00 + 24.61
Local + H.	428	359	787	33.98	58.98 + 24.33
IPD + H + IPD.	368	240	648	45.64	47.19 + 24.22 + 44.22
Exact + H + Exact.	352	241	593	50.25	61.00 + 24.61 + 60.21
Local + H + Local.	363	257	621	47.90	58.98 + 24.33 + 56.10
ARCPR (3 iter.)	310	226	536	55.03	9464

6.1.2 Instance 2

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	501	612	1113	–	9.10
IPD.	426	562	988	11.23	71.02
Exact.	430	504	934	16.08	83.12
Local.	426	540	966	13.21	76.04
Base + H.	422	436	858	22.91	9.10 + 16.19
IPD + H.	402	471	873	21.56	71.02 + 22.33
Exact + H.	404	395	799	28.21	83.12 + 27.42
Local + H.	402	423	825	25.88	76.04 + 26.40
IPD + H + IPD.	334	409	743	33.24	71.02 + 22.33 + 70.00
Exact + H + Exact.	310	322	632	43.22	83.12 + 27.42 + 83.10
Local + H + Local.	305	334	639	42.58	76.04 + 26.40 + 77.05
ARCPR (3 iter.)	288	312	600	46.09	9828

6.1.3 Instance 3

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	639	751	1390	–	10.15
IPD.	588	731	1319	5.11	63.45
Exact.	569	637	1206	13.24	78.31
Local.	557	666	1223	12.01	72.44
Base + H.	512	620	1132	18.56	10.15 + 20.45
IPD + H.	313	333	646	53.53	63.45 + 20.55
Exact + H.	314	297	611	56.04	78.31 + 24.12
Local + H.	305	305	610	56.11	72.44 + 23.00
IPD + H + IPD.	255	236	491	64.68	63.45 + 20.55 + 67.43
Exact + H + Exact.	251	205	456	67.19	78.31 + 24.12 + 80.31
Local + H + Local.	326	203	529	61.94	72.44 + 23.00 + 70.10
ARCPR (2 iter.)	310	200	510	63.31	8640

6.2 Exponential Distribution with $\lambda = 10$

6.2.1 Instance 4

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	1381	1112	2493	–	8.12
IPD.	1289	940	2229	10.59	72.26
Exact.	1288	906	2194	11.99	91.27
Local.	1268	926	2194	11.99	93.46
Base + H.	993	902	1895	23.98	8.12 + 20.48
IPD + H.	991	912	1909	23.42	72.26 + 20.58
Exact + H.	986	889	1874	24.83	91.27 + 23.03
Local + H.	986	889	1874	24.83	93.46 + 23.00
IPD + H + IPD.	878	770	1648	33.89	72.26 + 20.58 + 73.09
Exact + H + Exact.	884	747	1631	34.58	91.27 + 20.58 + 92.36
Local + H + Local.	874	768	1642	34.14	93.46 + 23.00 + 97.11
ARCPR (4 iter.)	836	646	1482	40.55	9475

6.2.2 Instance 5

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	1006	1105	2111	–	15.53
IPD.	879	993	1872	11.32	214.19
Exact.	893	929	1822	13.69	228.65
Local.	891	934	1825	13.55	220.29
Base + H.	980	1036	2016	4.50	15.53 + 15.03
IPD + H.	824	948	1772	15.06	214.19 + 16.24
Exact + H.	835	881	1716	18.71	228.65 + 18.41
Local + H.	835	886	1721	18.47	220.29 + 17.54
IPD + H + IPD.	745	811	1556	26.29	214.19 + 16.24 + 182.17
Exact + H + Exact.	739	720	1459	30.89	228.65 + 18.41 + 210.00
Local + H + Local.	759	755	1514	28.28	220.29 + 17.54 + 188.63
ARCPR (2 iter.)	620	706	1326	37.18	8118

6.2.3 Instance 6

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	2002	2110	4112	–	15.21
IPD.	1824	1858	3682	10.46	92.35
Exact.	1770	1898	3668	10.80	112.91
Local.	1802	1868	3670	10.75	99.21
Base + H.	2006	2012	4018	2.29	15.21 + 19.04
IPD + H.	1694	1666	3360	18.28	92.35 + 21.31
Exact + H.	1681	1661	3341	18.75	112.91 + 22.05
Local + H.	1653	1685	3338	18.82	99.21 + 20.15
IPD + H + IPD.	1690	1595	3285	20.11	92.35 + 21.31 + 90.89
Exact + H + Exact.	1568	1413	2981	27.50	112.91 + 22.05 + 109.61
Local + H + Local.	1527	1421	2948	28.31	99.21 + 20.15 + 95.88
ARCPR (3 iter.)	1180	1322	2502	39.15	10011

6.3 Truncated Normal Distribution with $\mu = 5, \sigma = 10$

6.3.1 Instance 7

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	946	952	1898	–	14.19
IPD.	819	881	1700	10.43	168.74
Exact.	818	825	1643	13.44	217.43
Local.	819	879	1698	10.54	201.22
Base + H.	612	780	1392	26.65	14.19 + 18.22
IPD + H.	614	746	1360	28.35	168.74 + 19.41
Exact + H.	612	701	1313	30.82	217.43 + 27.61
Local + H.	614	746	1360	28.35	201.22 + 25.87
IPD + H + IPD.	510	451	961	49.37	168.74 + 19.41 + 168.00
Exact + H + Exact.	515	426	941	50.42	217.43 + 27.61 + 200.17
Local + H + Local.	525	465	990	47.84	201.22 + 25.87 + 197.33
ARCPR (3 iter.)	388	412	800	57.85	12215

6.3.2 Instance 8

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	548	673	1221	–	13.31
IPD.	503	570	1073	12.12	39.44
Exact.	510	516	1026	15.97	61.10
Local.	508	539	1047	14.25	51.14
Base + H.	545	564	1109	9.17	13.31 + 16.17
IPD + H.	528	422	950	22.19	39.44 + 19.82
Exact + H.	533	384	917	24.89	61.10 + 25.00
Local + H.	533	400	933	23.58	51.14 + 24.33
IPD + H + IPD.	420	254	674	44.80	39.44 + 19.82 + 39.00
Exact + H + Exact.	549	335	884	27.60	61.10 + 25.00 + 60.18
Local + H + Local.	571	278	849	30.47	51.14 + 24.33 + 49.27
ARCPR (2 iter.)	480	235	715	41.44	9967

6.3.3 Instance 9

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	837	842	1679	–	15.11
IPD.	783	677	1460	13.04	62.32
Exact.	772	652	1424	15.19	71.12
Local.	770	654	1424	15.19	70.77
Base + H.	675	637	1312	21.85	15.11 + 16.29
IPD + H.	667	643	1310	21.97	62.32 + 17.21
Exact + H.	657	620	1277	23.94	71.12 + 17.98
Local + H.	657	620	1277	23.94	70.77 + 18.65
IPD + H + IPD.	550	393	943	43.84	62.32 + 17.21 + 58.81
Exact + H + Exact.	523	364	888	47.11	71.12 + 17.98 + 69.14
Local + H + Local.	523	364	888	47.11	70.77 + 18.65 + 67.00
ARCPR (3 iter.)	418	295	713	57.53	10708

6.4 Truncated Normal Distribution with $\mu = 10, \sigma = 5$

6.4.1 Instance 10

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	781	990	1948	–	14.91
IPD.	668	909	1577	19.05	53.35
Exact.	686	853	1539	21.00	67.45
Local.	689	871	1560	19.92	59.16
Base + H.	796	811	1607	17.50	14.91 + 16.45
IPD + H.	667	905	1572	19.30	53.35 + 18.26
Exact + H.	682	851	1533	21.30	67.45 + 21.44
Local + H.	688	867	1555	20.17	59.16 + 19.08
IPD + H + IPD.	468	605	1077	44.71	53.35 + 18.26 + 55.28
Exact + H + Exact.	516	561	1073	44.91	67.45 + 21.44 + 62.91
Local + H + Local.	525	578	1103	43.38	59.16 + 19.08 + 57.12
ARCPR (3 iter.)	465	530	995	48.92	7114

6.4.2 Instance 11

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Average Time (s)
Base (B)	967	1131	2098	–	14.70
IPD.	870	1011	1881	10.34	61.72
Exact.	837	984	1821	13.20	74.38
Local.	833	1000	1833	12.63	69.27
Base + H.	899	1017	1916	8.67	14.70 + 16.71
IPD + H.	735	1044	1779	15.20	61.72 + 18.22
Exact + H.	714	1017	1730	17.54	74.38 + 20.54
Local + H.	711	1032	1744	16.87	69.27 + 21.55
IPD + H + IPD.	572	716	1288	38.61	61.72 + 18.22 + 66.19
Exact + H + Exact.	533	711	1244	40.71	74.38 + 20.54 + 69.00
Local + H + Local.	560	735	1294	38.32	69.27 + 21.55 + 62.14
ARCPR (2 iter.)	527	653	1180	43.76	4752

6.4.3 Instance 12

Approach	Average Aircraft Delay	Average Crew Delay	Average Total Delay	Avg. % Improv. over B	Time (s)
Base (B)	1020	1071	2091	–	15.04
IPD.	877	978	1855	11.29	67.41
Exact.	881	900	1781	14.83	92.12
Local.	889	911	1800	13.92	71.15
Base + H.	901	973	1874	10.38	15.04 + 16.19
IPD + H.	769	914	1683	19.51	67.41 + 18.12
Exact + H.	769	842	1611	22.95	92.12 + 16.44
Local + H.	778	851	1629	22.09	71.15 + 16.13
IPD + H + IPD.	622	616	1238	40.79	67.41 + 16.54 + 18.12
Exact + H + Exact.	638	578	1216	41.84	92.12 + 16.44 + 85.77
Local + H + Local.	646	569	1215	41.89	71.15 + 16.13 + 70.32
ARCPR (4 iter.)	512	614	1126	46.15	10440

From the tables above, our results indicate that our scenario-based approach of embedding delay scenarios within the aircraft routing and crew pairing subproblems of the Exact and Local approaches (universally) provides an improvement of 2 – 3% over the IPD approach. Moreover, while the IPD provides an 11.44% improvement over the Base Case, the Exact approach yields a 14.92% improvement over the Base Case, while the Local approach provides a 13.78% improvement over the Base Case.

Table 1: Average relative improvements of each [Approach + H] over [Approach].

Instance	$\frac{([IPD]-[IPD+H])}{IPD} \times 100\%$	$\frac{([Exact]-[Exact+H])}{Exact} \times 100\%$	$\frac{([Local]-[Local+H])}{Local} \times 100\%$
1	20.92 (24.56)	21.12 (23.63)	21.14 (25.00)
2	11.64 (28.57)	14.45 (28.26)	14.60 (33.33)
3	51.02 (24.49)	49.34 (27.08)	50.12 (27.08)
4	14.36 (27.85)	14.59 (26.25)	14.31 (25.61)
5	5.34 (20.55)	5.82 (21.62)	5.70 (22.66)
6	8.75 (23.46)	8.91 (24.32)	9.05 (25.97)
7	20.00 (12.05)	20.09 (13.38)	19.91 (12.18)
8	11.46 (24.05)	10.62 (12.36)	10.89 (23.40)
9	10.27 (11.78)	10.32 (9.90)	10.32 (11.15)
10	0.32 (7.06)	0.39 (5.45)	0.32 (5.57)
11	5.42 (5.25)	5.00 (8.18)	4.86 (5.98)
12	9.27 (4.15)	9.55 (4.09)	9.50 (4.66)
Average	14.06 (17.81)	14.18 (17.04)	14.23 (18.55)

Table 1 records the relative improvement of each Approach + H over each Approach for delay cost. The corresponding relative improvement (i.e. reduction) in standard deviation is listed in brackets. The results indicate that the heuristic applied to the solution obtained in each Approach provides a universal improvement of approximately 14.06 – 14.23% over each Approach respectively.

Table 2: Average relative improvements of each [Approach + H + Approach] over [Approach + H].

Instance	$\frac{([IPD+H]-[IPD+H+IPD])}{IPD+H} \times 100\%$	$\frac{([Exact+H]-[Exact+H+Exact])}{Exact+H} \times 100\%$	$\frac{([Local+H]-[Local+H+Local])}{Local+H} \times 100\%$
1	19.50	21.77	21.09
2	14.89	20.90	22.55
3	23.99	25.37	13.28
4	13.67	12.97	12.66
5	12.19	14.98	12.03
6	2.23	10.78	11.68
7	29.34	28.33	27.21
8	29.05	3.60	9.00
9	28.02	30.46	30.46
10	31.49	30.01	29.07
11	27.60	28.09	25.80
12	26.44	24.52	25.41
Average	19.88	19.37	18.48

Table 2 records the relative improvement of each Approach + H + Approach over each Approach + H to determine the effectiveness of re-solving the integrated aircraft routing and crew pairing problem using the departure times specified by the heuristic re-timing algorithm (H). It may be observed that each Approach + H + Approach universally improves upon each Approach + H with an average improvement of between 18.48–19.88% across the three approaches. This illustrates that the heuristic re-timing (H) has the potential to adjust the departure times so as to take advantage of the total delay information for the network, and thus open up further opportunities for improvement upon re-solving the problem using these new departure times.

Table 3: Average relative improvements of the ARCPR approach over each Approach + H.

Instance	$\frac{([\text{IPD}+\text{H}]-[\text{ARCPR}])}{\text{IPD}+\text{H}} \times 100\%$	$\frac{([\text{Exact}+\text{H}]-[\text{ARCPR}])}{\text{Exact}+\text{H}} \times 100\%$	$\frac{([\text{Local}+\text{H}]-[\text{ARCPR}])}{\text{Local}+\text{H}} \times 100\%$
1	33.42	29.29	31.89
2	31.27	24.91	27.27
3	21.05	16.53	16.39
4	22.37	20.92	21.17
5	25.17	22.73	22.95
6	25.54	25.11	25.04
7	41.18	39.07	41.18
8	24.84	22.14	23.47
9	45.57	44.17	44.17
10	36.70	35.09	36.01
11	33.67	31.79	32.34
12	33.10	30.11	30.88
Average	28.76	26.30	27.14

Some of the improvements can be very large, in particular for instance 10, re-solving using the new departures times as specified by the IPD + H, Exact + H and Local + H approaches results in an average improvement of 31.49%, 30.01% and 29.07% respectively. Furthermore, it may be observed that the heuristic re-timing has the potential to reduce not only the average delay, but the standard deviation of delays across the 20 batches - with an average relative improvement of between 17.04 – 18.55%.

Table 3 records the relative improvement of the ARCPR approach over each Approach + H. It may be observed that the ARCPR approach provides a universal improvement over all the other Approach + H combinations; with an average improvement of around 26.30 – 28.76%. This demonstrates that although the aircraft scheduling decisions are only partially integrated within the subproblem of the ARCPR approach, there is significant potential for the generation of much higher quality solutions. One of the drawbacks however, is that the ARCPR approach requires significantly greater computational time in order to achieve these improvements.

Given the relatively short computation times required for each Approach + H + Approach, and the potential to obtain good quality solutions relatively quickly, we provide a comparison in Table 4 of the results obtained via Approach + H + Approach with the ARCPR approach.

Table 4: Relative improvements of the ARCPR approach over each Approach + H + Approach.

Instance	$\frac{([\text{IPD}+\text{H}+\text{IPD}]-[\text{ARCPR}])}{\text{IPD}+\text{H}+\text{IPD}} \times 100\%$	$\frac{([\text{Exact}+\text{H}+\text{Exact}]-[\text{ARCPR}])}{\text{Exact}+\text{H}+\text{Exact}} \times 100\%$	$\frac{([\text{Local}+\text{H}+\text{Local}]-[\text{ARCPR}])}{\text{Local}+\text{H}+\text{Local}} \times 100\%$
1	17.28	9.61	13.69
2	19.25	5.06	6.10
3	-3.87	-11.84	3.59
4	10.07	9.14	9.74
5	14.78	9.12	12.42
6	23.84	16.07	15.13
7	16.75	14.98	19.19
8	-5.93	19.23	15.90
9	24.39	19.71	19.71
10	7.61	7.27	9.79
11	8.39	5.14	8.81
12	9.05	7.40	7.33
Average	10.89	8.53	10.88

It may be observed in the table that the ARCPR approach provides the greatest improvement (approx. 11%) over the IPD + H + IPD and the Local + H + Local approaches, with an improvement of approximately 8.5% over the Exact + H + Exact approach.

It is interesting to note that, the ARCPR has the potential to provide a dramatic improvement over the **Approach + H + Approach**; for example, in Instance 9, it may be observed that the ARCPR provides a 24.39% improvement on average over the **IPD + H + IPD** approach. However, for a small number of cases (3/36) in Table 4, the **Approach + H + Approach** outperforms the ARCPR approach on average. This indicates that although the ARCPR approach benefits from the additional (partial) integration of the scheduling decisions within the aircraft subproblem, the ARCPR is still sub-optimal and may in some instances be outperformed by the **Approach + H + Approach**. Each **Approach** and the ARCPR Algorithm was able to be solved to optimality, and did not require the implementation of a branching scheme. For larger networks however, a scheme such as Ryan-Foster branching [2] may be applied to obtain the optimal integer solution for each instance.

7 Discussion

Our integrated scenario-based approaches for robust aircraft routing, crew pairing and re-timing possess many advantages over (i) existing re-timing approaches, which neglect delay propagation effects between aircraft and crew and (ii) traditional exact approaches, which have a tendency to suffer computationally, from an explosion of variables in the master problem.

Firstly, our results indicate that our scenario-based approach of embedding additional delay scenarios within the aircraft routing and crew pairing subproblems of the **Exact** and **Local** approaches (universally) provides an improvement of 2 – 3% in delay propagation over the **IPD** approach of [9], which uses only mean delay information. This indicates that including additional delay information within the aircraft routing and the crew pairing subproblems in the form of scenarios has the potential to improve the overall solution robustness. One of the limitations associated with the **Exact** approach is the potential for it to become computationally expensive for large networks; in such cases, the **Local** approach may be more computationally desirable.

Secondly, our proposed re-timing heuristic (**H**) when applied to each of these approaches, provided a clear and universal improvement of approximately 14% in average delay propagation reduction; with relatively little computational time (order of seconds) required to achieve such an improvement. One of the primary advantages of this proposed heuristic is that it (locally) re-times an incumbent aircraft routing and crew pairing assignment in such a way as to minimize delay propagation whilst preserving each of the aircraft routes and crew pairings. Moreover, it has the advantage of being able to be solved quickly; and so may allow an airline to improve the robustness of their solution without having to “undo” their original assignments.

Thirdly, we investigated whether it was possible to improve our solution further by re-solving the integrated aircraft routing and crew pairing problem using the new departure times as chosen by the heuristic. We observed that this resulted in a significant and universal improvement of around 18.5 – 20% over the result obtained using the heuristic (i.e. **Approach+H**). An advantage of this approach (i.e. **Approach+H+Approach**) is the ease with which it can be implemented and the possibility of achieving significant improvements without the need to insert potentially thousands of additional decision variables within the master problem. For this reason, these iterative approaches enable our methods to be applied to larger problems instances than fully integrated approaches.

Finally, our proposed ARCPR model integrating all three aspects of aircraft routing, crew pairing and re-timing into one model displayed the most promising results, with an even greater improvement of approximately 8.5 – 10.8% over each **Approach+H+Approach**, but with an accompanying increase in computational cost.

In the present work we have concentrated on reducing costs due to propagated delays,

however, our proposed methodology can also be used as an add-on to other string-based cost-minimization optimization problems.

Acknowledgements

The authors thank Irina Dumitrescu for helpful discussions on labeling algorithms. MD acknowledges support from the Australian Research Council (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] S. AhmadBeygi, A. Cohn, and M. Lapp. Decreasing Airline Delay Propagation by Re-Allocating Scheduled Slack. *IIE Transactions*, 42:478–489, 2010.
- [2] C. Barnhart, N. Boland, L. Clarke, E. Johnson, C. Nemhauser, and R. Shenoi. Flight String Models for Aircraft Fleeting and Routing. *Transportation Science*, 32:208–220, 1998b.
- [3] 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.
- [4] C. Barnhart, T. S. Kniker, and M. Lohatepanont. Itinerary-Based Airline Fleet Assignment. *Transportation Science*, 36(2):199–217, 2002.
- [5] N. Bélanger, G. Desaulniers, F. Soumis, and J. Desrosiers. Periodic Fleet Assignment with Time Windows, Spacing Constraints and Time Dependent Revenues. *European Journal of Operational Research*, 175:1754–1766, 2006.
- [6] G. Desaulniers, J. Desrosiers, Y. Dumas, M.M. Solomon, and F. Soumis. Daily Aircraft Routing and Scheduling. *Management Science*, 1997.
- [7] M. Desrochers and F. Soumis. A Generalized Permanent Labelling Algorithm for the Shortest Path Problem with Time Windows. *Informatics*, 26(3):191–212, 1988.
- [8] I. Dumitrescu and N. Boland. Improved preprocessing, labeling and scaling algorithms for the weight-constrained shortest path problem. *Networks*, 42(3):135–153, 2003.
- [9] M. Dunbar, G. Froyland, and C. L. Wu. Robust Airline Schedule Planning: Minimizing Propagated Delay in an Integrated Routing and Crewing Framework. *Transportation Science*, 46(2):204–216, 2012.
- [10] International Air Transport Association (IATA). Fact sheet: Economic and social benefits of air transport. http://www.iata.org/pressroom/facts_figures. Date Accessed: 3rd January, 2012.
- [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. *Planning for Robust Airline Operations: Optimizing Aircraft Routing and Flight Departure Times to Achieve Minimum Passenger Disruptions*. PhD thesis, 2003.

- [13] 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.
- [14] M. Lohatepanont and C. Barnhart. Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment. *Transportation Science*, 38(1), 2004.
- [15] A. Mercier and F. Soumis. An Integrated Aircraft Routing, Crew Scheduling and Flight Retiming Model. *Computers and Operations Research*, 34:2251–2265, 2007.
- [16] N. Papadakos. Integrated Airline Scheduling. *Computers and Operations Research*, 36:176–195, 2009.
- [17] Research and Innovative Technology Administration. Bureau of Transportation Statistics (BTS). <http://www.transtats.bts.gov/>. Date Accessed: August 10, 2010.
- [18] B. Rexing, C. Barnhart, T. Kniker, A. Jarrah, and N. Krishnamurthy. Airline Fleet Assignment with Time Windows. *Transportation Science*, 34(1):1–20, 2000.
- [19] S. Sarmadi. Minimizing Airline Passenger Delay through Integrated Flight Scheduling and Aircraft Routing. Master’s thesis, 2004. <http://hdl.handle.net/1721.1/29401>.
- [20] O. Weide. *Robust and Integrated Airline Scheduling*. PhD thesis, Department of Engineering Science, The University of Auckland, 2009. Chapter 6, 173-195.
- [21] O. Weide, D. Ryan, and M. Ehrgott. An Iterative Approach to Robust and Integrated Aircraft Routing and Crew Scheduling. *Computers and Operations Research*, 37:883–844, 2010.
- [22] C. L. Wu. Inherent Delays and Operational Reliability of Airline Schedules. *Journal of Air Transport Management*, 11(4):273–282, 2005.
- [23] C. L. Wu. Delay Propagation Modelling and the Implications In Robust Airline Scheduling. *Transportation Research Board 87th Annual Meeting*, 2008.
- [24] C. L. Wu. *Airline Operations and Delay Management—Insights from Airline Economics, Networks and Strategic Schedule Planning*. Ashgate, Farnham, UK, 2010.