# The daily tail assignment problem under operational uncertainty using look-ahead maintenance constraints

Stephen J Maher[*1,2], Guy Desaulniers[3], and François Soumis[3]

[1]Department of Optimization, Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany.
[2]Department of Management Science, Lancaster University, Bailrigg, Lancaster LA1 4YX, UK
[3]École Polytechnique de Montréal and GERAD, Department of Mathematics and Industrial Engineering, Montréal Québec H3C 3A7, Canada.

## Abstract

The tail assignment problem is a critical part of the airline planning process that assigns specific aircraft to sequences of flights, called lines-of-flight, to satisfy operational constraints. The aim of this paper is to develop an operationally flexible method, based upon the one-day routes business model, to compute tail assignments that satisfy short-range—within the next three days—aircraft maintenance requirements. While maintenance plans commonly span multiple days, the methods used to compute tail assignments for the given plans can be overly complex and provide little recourse in the event of schedule perturbations. The presented approach addresses operational uncertainty by using solutions from the one-day routes aircraft maintenance routing approach as input. The daily tail assignment problem is solved with an objective to satisfy maintenance requirements explicitly for the current day and implicitly for the subsequent two days. A computational study will be performed to assess the performance of exact and heuristic solution algorithms that modify the input lines-of-flight to reduce maintenance misalignments. The daily tail assignment problem and the developed algorithms are demonstrated to compute solutions that effectively satisfy maintenance requirements when evaluated using input data collected from three different airlines.

*Key words*: transportation, tail assignment, maintenance planning, branch-and-price, iterative algorithm

# 1    Introduction

The tail assignment problem (TAP) is a component of the airline planning process—separated by aircraft type—that involves the assignment of tasks to aircraft to satisfy operational constraints. A task within

---

*Corresponding author: s.maher3@lancaster.ac.uk

the TAP is defined as a sequence of flights that can be performed by a single aircraft, which is termed a line-of-flight (LOF). Operational constraints include: ensuring each flight is operated by an aircraft, satisfying aircraft maintenance requirements and planning through flights (matching high valued inbound and out-bound flights with the same aircraft).

The planning of LOFs typically occurs months in advance of operations. The planning stage involves solving an aircraft routing problem that constructs LOFs, which are expected to satisfy operational constraints. Fixing the LOFs in advance of operations is valuable for the construction of crew schedules— given the cost benefit of crew following aircraft—and organising ground services. Unfortunately, schedule perturbations are highly prevalent in daily operations. Such schedule perturbations can significantly impact the operational advantages derived from using fixed LOFs by causing infeasibilities in the planned aircraft routings and crew pairings.

Many different business models are employed by airlines for the construction and assignment of aircraft LOFs [14]. Traditional business models involve the construction of LOFs that span multiple days, in some cases up to a month. By spanning many days, such LOFs are highly susceptible to disruption. It is typical that many aircraft do not finish on their originally assigned LOFs. Also, at the end of an operational period, the performed LOFs may look vastly different to those constructed during the planning stage. Thus, constructing LOFs to span shorter time periods and then performing the aircraft assignment each night is expected to have significant advantages over the traditional business models in handling schedule perturbations.

A recent business model developed to address the disruption caused by schedule perturbations uses *one-day routes*—LOFs that span a single day and ensure the current days maintenance requirements are satisfied. The planning optimisation problem developed for the one-day routes business model is presented by Maher et al. [18]. In this paper, the suite of optimisation problems for this business model will be completed with the development of a complementary operational optimisation problem. This problem concerns the overnight assignment of aircraft to one-day routes LOFs to satisfy maintenance requirements. The maintenance requirements for the following two days are satisfied through the development of look-ahead maintenance constraints. Further, to address unavoidable schedule perturbations an adjustment procedure is employed that minimally modifies the input LOFs. A computational study will show the effectiveness of the one-day routes business model for satisfying daily maintenance requirements.

## 1.1 Literature review

Different business models for LOF construction in the aircraft routing problem (ARP)—*string*, *big-cycle* and *one-day routes* approaches—are reviewed by Lacasse-Guay et al. [14]. The string and big-cycle approaches are commonly solved over a number of days to provide a maintenance plan for the complete airline fleet. Alternatively, the one-day routes approach is solved to identify LOFs for all aircraft that

span only a single day. Maintenance planning is performed by only considering the aircraft requiring maintenance at the end of the current day. Alternative approaches draw upon and combine concepts from the three categories presented by Lacasse-Guay [14]. Such approaches construct aircraft routes that span time periods ranging from one day to multiple weeks without the restrictions of the string or big-cycle methods. It is important to note that all of these approaches can be solved many months in advance of the day of operations.

The string maintenance planning approach constructs a set of generic flight sequences, each to be performed by a single aircraft, that originate and terminate at maintenance opportunities. An individual flight sequence, termed a flight route, is constructed to be maintenance feasible. As such, the solution to a set partitioning problem, selecting a set of flight routes that covers all flights within a given time period, satisfies a fleet's maintenance requirements. Examples of the string maintenance planning approach are presented by Barnhart et al. [3] and Sriram and Haghani [21]. The big-cycle approach to maintenance planning involves the construction of a single route spanning multiple days that covers every flight in the schedule. Equal utilisation motivates this approach. This is achieved by constructing a single cycle that includes all flights to be operated by all aircraft. Examples of the big-cycle approach are presented by Feo and Bard [7], Clarke et al. [5], Gopalan and Talluri [10] and Talluri [22]. It is important to note that Feo and Bard [7], Gopalan and Talluri [10] and Talluri [22] construct big-cycle solutions using sets of LOFs that span a single day.

An approach that lies at the intersection of the string and big cycle approaches is the weekly aircraft maintenance routing problem presented by Liang and Chaovalitwongse [16]. The developed formulation is solved to provide a weekly rotation tour that satisfies aircraft maintenance requirements. An extension to the weekly aircraft maintenance routing problem presented by Liang, Feng et al. [17] aims to minimise delay propagation. The formulation of Liang, Feng et al. [17] draws upon concepts from one-day routes with the construction of single day LOFs that are concatenated to form weekly aircraft routing solutions.

The one-day routes approach is vastly different from the two previously discussed. This approach is applied to identify flight routes that span a single day. The objective of the one-day routes approach is to ensure that a sufficient number of flight routes from each airport terminate at a maintenance station so that the maintenance critical aircraft can receive maintenance that night. The one-day routes ARP is inherently stochastic since it assumes operations from previous days will perturb the maintenance plan. Examples of the one-day routes approach are presented by Heinhold [13], Lapp and Cohn [15] and Maher et al. [18].

While the ARP approaches presented above involve the generation of LOFs for input to the TAP, alternative methods have been proposed that combine the ARP and TAP. One of the most detailed investigations of the TAP developed in this manner is presented by Grönkvist [11]. The TAP proposed by Grönkvist [11] is solved a month at a time and comprises features from the fleet assignment, aircraft routing, maintenance planning and through assignment problems. The construction of LOFs within the

TAP to minimise a robustness measure is presented by Borndörfer et al. [4]. Additionally, Borndörfer et al. [4] construct LOFs to adjust for any perturbations from preceding days that may have affected the maintenance plan. Only an approximate solution is given by Borndörfer et al. [4], since the integer solution is given by a rounding heuristics applied to the optimal LP solution. Finally, flexibility in the construction of LOFs is achieved by Ruther [19] by considering the TAP as a component of an integrated airline planning problem. The problem presented by Ruther [19] is posed to be solved approximately four days before the day of operations to adjust the planned solutions in response to schedule perturbations.

A compact formulation for the single day maintenance routing problem is presented by Haouari et al. [12] that identifies a tail assignment solution without the explicit construction of LOFs. The authors develop a formulation that is polynomial in size and present effective solution methods based upon the reformulation-linearisation technique. Due to the deterministic focus of Haouari et al. [12], the presented formulation of the single day aircraft maintenance routing problem is vastly different to other one-day routes approaches by Heinhold [13], Lapp and Cohn [15] and Maher et al. [18]. Finally, Başdere and Bilge [2] presents a compact formulation for the aircraft maintenance routing problem with an objective to maximise the aircraft utilisation. Similar to Haouari et al. [12], LOFs are not explicitly constructed, but the routes are implicitly defined by the assignment of flight connections to aircraft.

Many variants of the TAP have been developed in response to different airline business practices and planning horizons. The longest planning horizon proposed for the TAP is a month, which is used by Grönkvist [11]; however, it is suggested that any planning period could be used. One week, or seven days, is a common planning horizon that guided the development of TAP variants presented by Başdere and Bilge [2], Liang and Chaovalitwongse [16], Liang, Feng et al. [17], Ruther [19], and Sriram and Haghani [21]. Regular planning horizons are not a requirement of the TAP and there are many instances where the horizon is determined by business practices or planning restrictions. Variants employing irregular planning horizons include Barnhart [3] with flight strings spanning a maintenance interval, Clarke et al. [5] with a big cycle that spans a number of days equal to the number of aircraft, Feo and Bard [7] where the time horizon is dictated by the inputs, and the three- and four-day maintenance planning problems by Gopalan and Talluri [10] and Talluri [22] respectively. The shortest planning horizon for the TAP is a single day. Variants of the TAP spanning a single day are presented by Börndorfer et al. [4], Froyland et al. [8], Haouari et al. [12], Heinhold [13] and Lapp and Cohn [15]. The focus of this paper is a variant of the TAP that assigns daily LOFs to aircraft. The developed approach aims to complement one-day route LOF planning approaches while providing an alternative to TAP variants formulated with time horizons that span multiple days.

In practice, the input LOFs are generally not suitable for satisfying aircraft operational requirements. In such cases, a process that modifies the input LOFs is required. One of the most common LOF adjustment processes, and possibly the most simple, is aircraft swapping [1]. By identifying two aircraft, one requiring maintenance that is assigned to an LOF not terminating at a maintenance base

and the other not requiring maintenance but assigned to an LOF terminating at a maintenance base, that are located at the same airport at the same time during the day, maintenance requirements can be satisfied by performing an aircraft swap. This approach is considered by Lapp and Cohn [15] by performing a simple splicing of two LOFs to improve maintenance reachability. While aircraft swapping is easily implemented in practice, a more sophisticated rerouting approach may be required to satisfy all maintenance requirements. A more involved approach that performs multiple flight changes for an aircraft LOF is presented in this paper.

Various solution approaches have been applied to solve the tail assignment and maintenance planning problems. Branch-and-price is popular for problem formulations where aircraft routes are not provided a priori. This is a feature of the TAP developed by Grönkvist [11], Borndörfer et al. [4] and Ruther [19] and the maintenance planning approaches by Barnhart et al. [3] and Maher et al. [18]. Alternatively, network based approaches are presented by Liang and Chaovalitwongse [16] and Liang, Feng et al. [17] that generate LOFs spanning from a single day up to a one week. There are many cases where the LOFs are generated by an ARP and provided as input to the TAP. For such problem formulations, solution methods including Lagrangian relaxation and subgradient algorithms [5], problem specific heuristics [2,7,10,21,22] or general-purpose mixed-integer programming solvers [2, 12, 15] have been employed. Observations suggest that exact solution algorithms, such as branch-and-price, can be overly time-consuming and not suitable for the practical implementation of algorithms. However, the high solution quality that is achievable using exact solution algorithms is desired.

A compromise between the exact solution algorithm of branch-and-price and problem-specific heuristics is presented in the form of iterative solution algorithms [6,23]. Iterative algorithms have previously been employed to solve integrated airline planning problems, whereby the solution to one stage can be fixed prior to solving the alternate stage and then iterating between the two problems. This paper aims to extend this technique with the development of an iterative solution algorithm to improve the solution of the TAP using a set of input LOFs. The algorithm involves i) solving the TAP to identify any infeasibilities in the maintenance plan, and ii) solving the TAP using branch-and-price to generate flight routes for a subset of aircraft to address these infeasibilities. The algorithm executes in run times that are better or competitive with exact approaches while still achieving high-quality solutions.

## 1.2   Contributions

The mathematical model developed in this paper draws upon many concepts from the literature. The assignment of fixed LOFs to aircraft that are adjusted to satisfy maintenance requirements is presented by Gopalan and Talluri [10] and Talluri [22]. Focusing on the work of Talluri [22], the main limitations are the requirement of an Euler tour and the use of heuristics to repair maintenance infeasibilities. In particular, the Euler tour requirement is overly restrictive. This paper significantly extends the work of Talluri [22] with the development of a TAP that is flexible in regards to the input schedule and

the maintenance requirements of the aircraft. Without the construction of an Euler tour and by only providing an assignment for a single day the solution of the TAP developed in this paper provides a maintenance plan that is robust to schedule perturbations. Further, the developed solution approaches for the TAP are exact and heuristic in nature. As such, provable optimality bounds are available for the considered problem instances.

In contrast to the work of Gopalan and Talluri [10] and Talluri [22], fully flexible approaches combining the aircraft routing and tail assignment problems are presented by Grönkvist [11], Börndorfer et al. [4] and Ruther [19]. Flexibility in the TAP allows the generation of aircraft routes for the tail assignment to satisfy operational constraints, such as maintenance requirements. However, maintenance planning beyond the day-of-operations is not considered. The TAP presented in this paper provides both full planning flexibility and a heuristic approach for practical application. Additionally, look-ahead maintenance constraints have been developed to complement LOF construction approaches by providing an effective method to implicitly satisfy future maintenance requirements.

The contributions of this paper focus on the analysis of the new business model employing the one-day routes approach. The contributions include

- the development of a tail assignment problem using one-day routes as input that is solved each day—the daily tail assignment problem (DTAP)

- the evaluation of the use of one-day routes, as produced by the approaches of Heinhold [13], Lapp and Cohn [15] and Maher et al. [18], as a fixed input for the DTAP,

- the formulation of look-ahead constraints that implicitly satisfy day-two and day-three maintenance requirements,

- the presentation of a branch-and-price algorithm that reconstructs LOFs to improve the maintenance planning achieved by the DTAP,

- the development of an iterative algorithm that reduces the computational effort of the LOFs reconstruction process and

- a detailed computational study demonstrating the strengths and weaknesses of solution approaches for the DTAP.

## 1.3  Paper structure

The problem description and formulation is presented in Section 2. The discussion in Section 2 involves two parts, the first presenting the DTAP to satisfy maintenance requirements for day one using one-day routes as input and the second introducing the look-ahead constraints that implicitly satisfy day-two and day-three maintenance requirements. Two different solution algorithms are presented in Section 3—an

exact branch-and-price algorithm and a heuristic iterative algorithm. The branch-and-price algorithm presented in Section 3.1 resolves the DTAP, using the input LOFs as a starting solution, to eliminate the unsatisfied maintenance requirements arising from solving the DTAP using a fixed set of input LOFs. An iterative algorithm is developed in Section 3.2 that aims to improve upon the computational performance of the exact branch-and-price algorithm. Section 4 describes the data used to evaluate the DTAP developed in this paper. The computational results involving various flight schedules are presented in Section 5. Finally, Section 6 provides some concluding comments.

## 2   The daily tail assignment problem

The DTAP is solved immediately prior to the day of operations to aid the recovery of planned assignments that are disrupted as a result of schedule perturbations. Two critical features of aircraft routing that are highly susceptible to schedule perturbations are addressed by the DTAP: the assignment of LOFs to aircraft and maintenance requirements. The DTAP is solved at an arbitrary time when it is expected that most aircraft are located on the ground. The fleets considered for the DTAP are short-haul and medium-haul fleets, hence there are no overnight flights. However, airlines may operate in multiple time zones. Thus, all flights departing between midnight and midnight (local time) on consecutive days are considered to belong to the same one-day schedule. It is possible for aircraft to be operating flights while the DTAP is being solved, requiring an estimated arrival time to be used in this problem for such aircraft. Finally, to satisfy regulatory requirements aircraft are expected to receive maintenance once every six days.

This paper presents the DTAP defined as: Given a set of input LOFs that span the next day of operations, assign to each aircraft exactly one LOF that originates from the current aircraft location. The LOF assignment minimises the number of day-one maintenance-critical aircraft *not* terminating at a maintenance base at the end of the next day. A further aspect unique to this paper is the consideration of day-two and day-three maintenance requirements. Two additional sets of LOFs—one each for days two and three—that span a single day are provided as input. All three sets of input LOFs are not required to be identical. The number of available maintenance routes from the end of day one to the end of days two and three are computed from the day-two and day-three LOFs. Using the number of available maintenance routes as input, the DTAP minimises—as a secondary objective—the number of day-two and day-three maintenance critical aircraft unable to receive maintenance on the respective days without the explicit assignment of LOFs.

Explicit maintenance planning is only modelled in the DTAP for the forthcoming day of operation. However, aircraft requiring maintenance on days two and three are still considered. This is supported by the addition of constraints that ensure a sufficient number of maintenance routes depart from each overnight airport on day two and day three for the maintenance critical aircraft. This implicit considera-

tion of maintenance requirements is a novel approach that has not been previously investigated. For ease of exposition Section 2.1 presents the DTAP that only considers the day-one maintenance planning. The unique implicit enforcement of the day-two and day-three maintenance requirements using look-ahead maintenance constraints is presented in Section 2.2.

The above problem description considers the case where the LOFs are provided as input to the DTAP. This is common in practice, since the LOFs from the aircraft routing solution are typically provided as input. It is also common that as a result of schedule perturbation the aircraft routing LOFs do not provide sufficient opportunities to satisfy maintenance requirements. Thereby, modifications to the LOFs are necessary to satisfy all operational requirements. In this paper, the discussion of the solution algorithm involves three parts. First, Sections 2.1 and 2.2 present the basic model where the LOFs are provided as input. Second, the modification of the mathematical model to permit changes to the input LOFs is presented in Section 2.3. Finally, the solution algorithms developed to respond to schedule perturbations are described in Section 3.

## 2.1  The daily tail assignment problem without look-ahead constraints

There exist three key components of the DTAP, namely the overnight airports, aircraft and the LOFs. Let $B$ be the set of overnight airports. A set of aircraft $R^b$, all of the same type, are located at each overnight airport $b \in B$ to commence the forthcoming day of operation. The LOFs available for assignment to an aircraft $r \in R^b$ are given by the set $P^r$, which is indexed by $p$. $P^r$ is populated with LOFs that are identical for all $r \in R^b$ originating for the same overnight airport $b$.

The LOF assignment is modelled using the binary variables $y_p^r$ that equal one if aircraft $r$ is assigned to LOF $p$, and zero otherwise. The origination and termination locations of the LOF and the contained flights are directly considered in the DTAP model. All flights of an airline schedule, which belong to the set denoted by $N$, are each included in exactly one input LOF. The parameter $a_{fp}$ is defined to equal one if flight $f$ is included in LOF $p$. Each LOF $p$ originates and terminates at an overnight airport. A subset of overnight airports, $\hat{B} \subset B$, are identified as maintenance stations.

The day-one maintenance requirements are addressed by attempting to assign each maintenance critical aircraft to an LOF terminating at a maintenance station. The parameters $o_p$ are defined to equal one if the LOF $p$ terminates at a maintenance station, and zero otherwise. A given aircraft $r \in R^b, b \in B$ is identified as maintenance critical—defined as requiring maintenance at the end of the current day—by the parameter $\theta_1^r = 1$. Otherwise $\theta_1^r = 0$. Since disruptions from preceding days may prohibit aircraft from entering a maintenance station, the slack variables $s_1^r$, $r \in R^b$, $b \in B$, are introduced, with the objective coefficient $c_1^r$, to penalise any infeasibility of the maintenance plan.

The DTAP without look-ahead maintenance constraints can be modelled as the following mixed

integer program:

$$\min \quad \sum_{b \in B} \sum_{r \in R^b} c_1^r s_1^r, \tag{1}$$

$$\text{s.t.} \quad \sum_{b \in B} \sum_{r \in R^b} \sum_{p \in P^r} a_{fp} y_p^r = 1 \quad \forall f \in N, \tag{2}$$

$$\sum_{p \in P^r} y_p^r \leq 1 \quad \forall b \in B, \forall r \in R^b, \tag{3}$$

$$\sum_{p \in P^r} o_p y_p^r + s_1^r \geq \theta_1^r \quad \forall b \in B, \forall r \in R^b, \tag{4}$$

$$y_p^r \in \{0, 1\} \quad \forall r \in R, \forall p \in P, \tag{5}$$

$$s_1^r \geq 0 \quad \forall r \in R. \tag{6}$$

The problem defined by (1)-(6) assigns LOF to aircraft and minimises the violation of maintenance requirements. While it is possible to formulate the tail assignment as a set partitioning of input LOFs, it is more convenient to formulate this problem as a set partitioning of flights in the network. The latter formulation aids the development of a re-optimisation method that modifies LOFs when (1)-(6) is maintenance infeasible. The proposed re-optimisation method is presented in Section 3. The set partitioning of flights is given by constraints (2) that ensure every flight $f \in N$ is assigned to exactly one aircraft. Each aircraft must operate exactly one LOF, which is given by constraints (3). The day-one maintenance requirements for the maintenance critical aircraft are enforced with constraints (4). These constraints include a slack variable $s_1^r$ to penalise any maintenance violations.

## 2.2 Modelling the look-ahead maintenance constraints

The termination locations of each aircraft can be used by the look-ahead maintenance constraints to implicitly satisfy day-two and day-three maintenance requirements. In particular, the maintenance LOFs and maintenance paths, Definitions 2.2.1 and 2.2.2 respectively, are used to implicitly assign LOFs to aircraft. Since the look-ahead constraints rely on LOFs originating from overnight airports on different days, for convenience $b_i$ is used to denote overnight airport $b$ at the start of day $i$.

**Definition 2.2.1** (Maintenance LOF). *A maintenance LOF (MLOF) is an LOF that terminates at a maintenance station at the end of the day.*

**Definition 2.2.2** (Maintenance path). *A maintenance path is the concatenation of an LOF from day $i$ and an MLOF from day $i + 1$, where the origination overnight airport for the MLOF, $b_{i+1}$, is identical to the termination overnight airport for the LOF.*

As a result of schedule perturbations, the number of maintenance critical aircraft at each overnight airport is uncertain. Additionally, in the response to disruptions, LOFs are modified before and during the day of operation. As such, the number of MLOFs departing from each overnight airport is unknown.

Thus, by focusing only on the identification of termination locations and the number of MLOFs, stochastic and robust approaches can be effectively employed. While robust formulations of this problem are possible, this paper will focus only on the deterministic form.

The key parameters for the maintenance look-ahead constraints are the number of LOF and MLOFs from each airport and the number of maintenance paths. The number of LOFs originating from overnight airport $i$ and terminating at overnight airport $j$ is given by $L_{ij}$. The parameter $M_{b_i}$ is the number of LOFs that can be used to satisfy the maintenance requirements for maintenance critical aircraft originating from overnight airport $b_i$, i.e., the number of MLOFs. A maintenance critical aircraft at overnight airport $b_i$ can only use MLOFs originating from $b_i$. Hence, the number of maintenance critical aircraft at $b_i$ that can receive maintenance at the end of day $i$ is bounded by $M_{b_i}$.

An example of the number of LOFs from each overnight airport is given in Figure 1a. The first level of the tree in Figure 1a are all LOFs departing from SYD on day 2. The second level shows all LOFs departing from BNE, MEL and SYD on day 3. For convenience, we will focus only on the LOFs departing from SYD. On day two, Figure 1a shows that there are 7 LOFs departing SYD. Since MEL is the only maintenance station, there only exists 3 MLOFs departing from SYD, which means $M_{SYD_2} = 3$. Similarly, on day three the total number of LOFs departing from SYD is 7 and the number of MLOFs is 3. As such, the parameter $M_{SYD_3}$ is set to 3. It can also be observed in Figure 1a that $M_{BNE_3} = 1$ and $M_{MEL_3} = 3$.

Given two overnight airports, one at the start of day $i$ and one at the start of day $i+1$, denoted by $b_i$ and $b_{i+1}$ respectively, the *number of maintenance paths* constructed by concatenating an LOF departing from $b_i$ and an MLOF departing from $b_{i+1}$ is given by

$$N_{b_i b_{i+1}} = \min\{L_{b_i b_{i+1}}, M_{b_{i+1}}\}. \tag{7}$$

The parameters $N_{b_2 b_3}$ identify any routing bottlenecks that can prevent the day-three maintenance critical aircraft receiving maintenance as required. As demonstrated in the example in Figure 1c, $N_{SYD,BNE} = 1$ (indicating a bottleneck on day three), $N_{SYD,MEL} = 3$ (indicating no bottleneck) and $N_{SYD,SYD} = 2$ (indicating a bottleneck on day two).
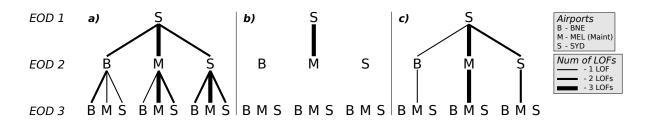


Figure 1: Available LOFs and maintenance routes from SYD at the end of day 1. a) All available LOFs. b) Routes from SYD terminating at the maintenance station (MEL) at end of day two. c) Routes from SYD terminating at the maintenance station (MEL) at the end of day three.

**Day-two maintenance requirements**    The day-two maintenance requirements are satisfied by simply counting the number of maintenance critical aircraft terminating at each airport at the end of day one. The parameter $\theta_2^r$ equals one if aircraft $r$ requires maintenance at the end of day two, and zero otherwise. The total number of maintenance routes departing from airport $b$ on day two is given by the parameter $M_b^2$. The difference between the number of day-two maintenance critical aircraft and $M_b^2$ indicates the feasibility of the day-one tail assignment. Any over demand for maintenance routes on day two is minimised with a set of penalty terms in the objective function, given by the variables $s_2^{b_2}$, $b_2 \in \hat{B}$. The termination location of LOF $p$ at the end of day one is given by the parameter $term(p)$. The addition of the following constraints evaluate the satisfaction of maintenance requirements for the day-two maintenance critical aircraft:

$$\sum_{b_1 \in B} \sum_{r \in R^{b_1}} \sum_{\substack{p \in P^{b_1} \mid \\ term(p)=b_2}} \theta_2^r y_p^r - s_2^{b_2} \leq M_{b_2}^2 \quad \forall b_2 \in B, \tag{8}$$

$$s_2^{b_2} \geq 0 \quad \forall b_2 \in B. \tag{9}$$

Constraints (8) count the number of day-two maintenance critical aircraft located at $b_2$ and sets the slack variable $s_2^{b_2}$ to penalise any maintenance misalignments.

**Day-three maintenance requirements**    Satisfying the day-three maintenance requirements involves counting the number of day-three maintenance critical aircraft and assigning a day-two termination location to each. The parameters $\theta_3^r$ equal one to identify whether aircraft $r$ requires maintenance at the end of day three, and zero otherwise. The variables $\xi_{b_2 b_3}$ count the number of day-three maintenance critical aircraft located at $b_2$ arriving at airport $b_3$ at the end of day two. Similar to days one and two, the number of maintenance routes departing from overnight airport $b_3$ is given by the parameter $M_{b_3}^3$. The additional constraints to evaluate maintenance feasibility for the day-three maintenance critical aircraft are given by

$$\sum_{b_1 \in B} \sum_{r \in R^{b_1}} \sum_{\substack{p \in P^{b_1} \mid \\ term(p)=b_2}} \theta_3^r y_p^r - s_3^{b_2} = \sum_{b_3 \in B} \xi_{b_2 b_3} \quad \forall b_2 \in B, \tag{10}$$

$$\sum_{b_3 \in \hat{B}} \xi_{b_2 b_3} + \sum_{b_1 \in B} \sum_{r \in R^{b_1}} \sum_{\substack{p \in P^{b_1} \mid \\ term(p)=b_2}} \theta_2^r y_p^r - s_2^{b_2} \leq M_{b_2}^2 \quad \forall b_2 \in B, \tag{11}$$

$$\sum_{b_2 \in B} \xi_{b_2 b_3} \leq M_{b_3}^3 \quad \forall b_3 \in B, \tag{12}$$

$$\xi_{b_2 b_3} \in [0, N^{b_2 b_3}] \quad \forall b_2, b_3 \in B, \tag{13}$$

$$s_3^{b_2} \geq 0 \quad \forall b_2 \in B. \tag{14}$$

Constraints (10) count the number of day-three maintenance critical aircraft located at $b_2$ at the end of day one. This constraint also assigns each of the maintenance critical aircraft to a maintenance

route from $b_2$ passing through $b_3$ using the variables $\xi_{b_2 b_3}$. Since the LOFs are provided as input, it is possible that the number of day-three maintenance critical aircraft arriving at $b_2$ is greater than $N^{b_2 b_3}$—causing an infeasibility. The slack variable $s_3^{b_2}$ is introduced to measure the extent of this infeasibility, which is penalised in the objective function. Setting $\xi_{b_2 b_3}$ by constraint (10) determines the number of maintenance critical aircraft originating from $b_3$ at the start of day three. Since day-two maintenance critical aircraft require a maintenance route departing from $b_2$, this reduces the number of maintenance routes passing through $b_3$ that are available for day-three maintenance critical aircraft, where $b_3 \in \hat{B}$. Hence, (11) constrains the number of day-two and day-three maintenance critical aircraft terminating at maintenance bases at the end of day two to at most $M_{b_2}^2$. To ensure that at most $N^{b_2 b_3}$ day-three maintenance critical aircraft arrive at $b_3$ from $b_2$ an upper bound is imposed on the variables $\xi_{b_2 b_3}$ as indicated by constraints (13).

Including both the day-two and day-three look-ahead maintenance constraints introduces dominated inequalities. Specifically, constraints (8) are completely dominated by constraints (11). As such, only constraints (11) are required in the implementation of the DTAP with day-two and day-three maintenance look-ahead.

The addition of constraints (9) and (10)-(14) to the DTAP requires the modification of the objective function (1). This modification involves adding for each overnight airport $b \in B$ the slack variables $s_2^b$ and $s_3^b$, which count the number of day-two and day-three maintenance misalignments, along with the cost parameters $c_2^b$ and $c_3^b$ respectively. The objective function used for the DTAP with look-ahead maintenance constraints is given by

$$\sum_{b \in B} \sum_{r \in R^b} c_1^r s_1^r + \sum_{b \in B} \left\{ c_2^b s_2^b + c_3^b s_3^b \right\}. \tag{15}$$

## 2.3   Responding to schedule perturbations

Schedule perturbations regularly affect daily operations and as a result aircraft terminate at locations different to what is planned. To address the impact of schedule perturbations, the operational model of the DTAP requires a set of LOFs, in addition to the input LOFs, that can reduce the number of maintenance misalignments. The original formulation of the DTAP—with and without look-ahead maintenance constraints—considers only the input LOFs, which are contained in the sets $P^r, r \in R^b, b \in B$. In response to schedule perturbations, the DTAP is solved with the additional sets $\bar{P}^r, r \in R^b, b \in B$ that describe all possible LOFs, not including the input LOFs, for the given flight schedule.

It is important to ensure that any LOF selected from $\bar{P}^r, r \in R^b, b \in B$ does not exhibit large deviation from the input LOFs. To achieve this, the set $\bar{C}$ is defined to contain all connections $(i, j)$ that are used by the input LOFs. Additionally, the cost term

$$c_p = \sum_{(i,j) \in p \backslash \bar{C}} c_{ij}, \tag{16}$$

is defined for all LOFs $p \in \bar{P}^r, r \in R^b, b \in B$, where $c_{ij}$ is a positive weight to penalise the use of each connection $(i, j)$ in $p$ that is not used by any input LOFs. The DTAP to respond to schedule perturbations is then given by replacing $P^r$ with $P^r \cup \bar{P}^r$ in constraints (2)-(14) and adding the term

$$\sum_{b \in B} \sum_{r \in R^b} \sum_{p \in \bar{P}^r} c_p y_p^r \tag{17}$$

to objective function (15).

The set of all possible LOFs is prohibitively large to directly solve the DTAP as a mixed integer program. As such, a branch-and-price approach is developed to dynamically construct and add LOFs to the DTAP. The details of exact and heuristic branch-and-price approaches employed to solve the operational model of the DTAP are described in Section 3.

# 3    Solution algorithms

An exact and a heuristic algorithm that dynamically generate LOFs and find solutions better than those obtained with fixed LOFs are developed. Both algorithms are based on column generation, where the heuristic algorithm generates LOFs only for a subset of aircraft. The exact branch-and-price algorithm is presented in Section 3.1 and the heuristic iterative algorithm is described in Section 3.2.

In practical applications of the DTAP, it is most critical to ensure that aircraft requiring maintenance on day one must be assigned an LOF terminating at a maintenance base. As such, the route adjustment process focuses on the modification of day-one LOFs. Through the use of look-ahead maintenance constraints, the adjustment of day-one LOFs will also reduce maintenance misalignments on subsequent days. It is possible to additionally modify day-two and day-three LOFs. However, this work would be redundant as it is expected that further route adjustments will be required at the start of each subsequent day in response to schedule perturbations—thereby the modified LOFs would require further modifications. As such, the solution algorithms presented here modify only the day-one input LOFs.

While the input LOFs are not expected to satisfy all operational constraints, their use is valuable from an algorithmic point of view. First, the aircraft operational requirements are typically similar on the day of operations compared to when the aircraft routing problem is solved. As such, only a small number of modifications to the LOFs should be required in any LOF adjustment process. Second, the input LOFs provide a good initial solution for the DTAP. It is expected that by using a good initial solution, the runtime required for the LOF adjustment procedure should be greatly reduced. The benefit from using a good initial solution will be shown in Section 5.2.2

## 3.1    Branch-and-price

Branch-and-price is a solution technique commonly used to solve aircraft routing and tail assignment problems. A branch-and-price algorithm uses column generation to solve the LP relaxation at every node

in the branch-and-bound tree. Special branching rules, such as the Ryan-Foster branching scheme [20], are required by this algorithm to ensure that the subproblem structure is not destroyed by branching decisions. While branch-and-price is a commonly applied solution technique, its use to solve the DTAP deviates from the traditional sequential solution approach where modification to the input LOFs is traditionally forbidden. This restriction arises because the modification of LOFs can have an impact on many of the interrelated airline resources, most importantly crew.

The branch-and-price algorithm solves the DTAP by dynamically generating LOFs that minimise the number of day-one maintenance misalignments and number of expected misalignments on day-two and day-three. This is achieved by defining the restricted master problem (RMP) as a minimisation problem with the objective function (15) and constraints given by (2)-(14). Additionally, only the LOFs contained in $P^r, r \in R^b, b \in B$ are initially included in the RMP. The branch-and-price algorithm dynamically generates LOFs from the sets $\bar{P}^r, r \in R^b, b \in B$.

### 3.1.1  Column generation subproblem

A column generation subproblem is formed for each aircraft $r \in R^b, b \in B$. The objective of each subproblem is to identify the aircraft routing variable with the minimum reduced cost. To facilitate the description of the column generation subproblem, the dual variables related to the constraints of the DTAP that appear in aircraft route variables reduced cost function will be presented. The dual variables for the flight coverage constraints (2) are defined as $\boldsymbol{\rho} = \{\rho_j, \forall j \in N\}$. For the LOF assignment constraints (3), the dual variables are defined as $\boldsymbol{\delta} = \{\delta_b^r, \forall b \in B, \forall r \in R^b\}$. The dual variables for the day-one maintenance enforcement constraints (4) are defined as $\boldsymbol{\alpha} = \{\alpha_b^r, \forall b \in B, \forall r \in R^b\}$. For the day-two maintenance enforcement constraints (11), the dual variables are defined as $\boldsymbol{\beta} = \{\beta_b, \forall b \in B\}$. Finally, the dual variables for the day-three maintenance critical count constraints (10) are defined as $\boldsymbol{\gamma} = \{\gamma_b, \forall b \in B\}$.

The column generation subproblem is a shortest path problem: Identifying a minimum cost path through a network from a single source to one of multiple sink nodes. The network is defined by a set of nodes given by $N$ and a set of edges given by the feasible connections between the flights contained in $N$. A connection between flights $i$ and $j$ contained in $N$, $(i, j)$, is deemed feasible if i) the destination of $i$ is the same as the origin of $j$, and ii) the departure time of $j$ occurs after the minimum turn time following the arrival of $i$. All feasible connections are contained in the set $C$. To describe the minimum cost path, the binary variables $w_{ij}^r$ equal one to indicate aircraft $r$ uses connection $(i, j)$ or zero otherwise. The objective coefficient of $w_{ij}^r$, denoted by $c_{ij}$, equals one if connection $(i, j)$ is not observed in any input LOFs and zero otherwise. Aircraft must originate from an overnight airport $b$ and may terminate at any overnight airport $b' \in B$, describing the source and sink nodes respectively. The binary parameters $\bar{o}_b$ are introduced to indicate whether maintenance can be performed at overnight airport $b$. Using these

definitions, the column generation subproblem is given by

$$C^r = \min \quad \sum_{(i,j)\in C} c_{ij} w_{ij}^r - \sum_{i\in N\cup B} \sum_{j\in N} \rho_j w_{ij}^r - \delta_b^r - \sum_{i\in N} \sum_{b'\in B} w_{ib'}^r \left\{ \bar{o}_{b'} \alpha_{b'}^r + \theta_2^r \beta_{b'} + \theta_3^r \gamma_{b'} \right\}, \quad (18)$$

$$\text{s.t.} \quad \sum_{i\in N} w_{ij}^r - \sum_{k\in N} w_{jk}^r = 0 \quad \forall j \in N, \quad (19)$$

$$\sum_{j\in N} w_{bj}^r = 1, \quad (20)$$

$$\sum_{j\in N} \sum_{b'\in B} w_{jb'}^r = 1, \quad (21)$$

$$w_{ij}^r \in \{0,1\} \quad \forall (i,j) \in C. \quad (22)$$

The objective function (18) is the reduced cost function of the routing variables for aircraft $r \in R^b, b \in B$. The flow balance at each node (flight) in the network is maintained by (19). The origin and destination of a flight route is enforced through constraints (20) and (21) respectively.

A consideration of the LOF reconstruction is the impact that the alternative connections contained in the newly generated routes has on crew. A feasible connection for a crew requires a minimum sit time, generally longer than the minimum turn time for aircraft, between the arrival of $i$ and the departure of $j$. However, crew may use the connection $(i,j)$ with a ground time less than the minimum sit time but greater than the minimum turn time if an aircraft also uses this connection. Such connections are called *short connections*. These connections are important for the crew scheduling solution and must be protected in the generation of aircraft routes. This is achieved in the column generation subproblem by ensuring that any short connections existing in the input LOFs are used in any solution to the DTAP.

### 3.1.2 Integer optimality

Two different branching rules are implemented to derive integer solutions and prove optimality—both variations of the Ryan-Foster branching scheme [20]. The first rule implemented is similar to the follow-on branching described by Froyland et al. [8]. Consider two flights $i$ and $j$ with the connection $(i,j)$. From the values of the variables $y_p^r, r \in R, p \in P^r \cup \bar{P}^r$ in the LP relaxation solution, it is possible to deduce the values of the connection variables $x_{ij}^r$ in the compact formulation. If $\sum_{r\in R} x_{ij}^r$ is fractional, then a branching is performed by enforcing the use of connections $(i,j)$ on the left branch if $i$ or $j$ are present in the generated flight routes and forbidding the use of the connection $(i,j)$ on the right branch. The second branching rule is a variation of the Ryan-Foster branching that enforces or forbids the use of individual flights by particular aircraft. Given flight $i$ and aircraft $r$, if $\sum_{j\in N} x_{ij}^r$ is fractional then the left branch enforces the use of exactly one copy of flight $i$ in all routes generated for aircraft $r$ and on the right branch all routes generated for aircraft $r$ must not contain flight $i$. In the implementation of branch-and-price for the DTAP, the follow-on branching is performed with a higher priority than the aircraft/flight branching rule.

## 3.2 Iterative algorithm

The iterative algorithm for the LOF adjustment process involves *evaluation* and *update* stages. The *evaluation* stage determines whether the maintenance plan can be satisfied with a fixed set of LOFs by directly solving the DTAP—using a general purpose MIP solver. The *update* stage identifies alternative day-one LOFs for a subset of aircraft. This stage employs a branch-and-price algorithm.

### 3.2.1 Identify aircraft for route adjustment

The solution to the evaluation stage is used to identify a subset of aircraft that require the generation of LOFs in the update stage. The aircraft selection is based upon maintenance requirements and the assignment of LOFs. The set of selected aircraft is given by $\hat{R}$. Only a subset of flights are used to generate LOFs in the route adjustment problem. This subset of flights, denoted by $\hat{N}$, is given by those appearing in LOFs assigned in the evaluation stage to aircraft in $\hat{R}$. Throughout the algorithm, updates to $\hat{R}$ induce updates to $\hat{N}$.

Following the first execution of the evaluation stage two different types of aircraft are identified for inclusion in $\hat{R}$. The first are day-one maintenance critical aircraft assigned to an LOF that does not terminate at a maintenance station. The second are the aircraft assigned LOFs terminating at maintenance stations that do not require maintenance at the end of day one.

Subsequent iterations of the algorithm augment $\hat{R}$ using the solution to the evaluation stage. The augmentation involves identifying intersecting LOFs. Two LOFs intersect if there exists a flight in one LOF that departs within an intersection window commencing after the arrival of a flight in the other LOF at the same airport. The intersection window has a duration that is given by the sum of the minimum time aircraft require between two connecting flights, called the turn time, and a small buffer. Aircraft are selected for inclusion in $\hat{R}$ if they are assigned LOFs intersecting with LOFs assigned to aircraft in $\hat{R}$.

### 3.2.2 Route adjustment problem

The main focus of the route adjustment problem (RAP) is to identify day-one LOFs for aircraft in $\hat{R}$ that minimise the maintenance misalignments for the whole fleet. Thus, the restriction of the aircraft in $\hat{R}$ to a fixed set of LOFs is relaxed. The LOFs provided as input for the DTAP are used to define the initial set of variables for the RAP. The variables, or columns, defined by the input LOFs are a subset of all possible aircraft routes for the given flight schedule. Hence, this problem represents the RMP for the branch-and-price algorithm used to solve the RAP.

The column generation subproblem for the RAP is identical to that presented in Section 3.1.1. One goal of the iterative algorithm is to reduce the computational time while still decreasing maintenance misalignments compared to the solution obtained with fixed LOFs. As such, restrictions are imposed on

the sets of aircraft, flights and connections used to define the column generation subproblem. In particular, a column generation subproblem is only formed for each aircraft $r \in R^b \cap \hat{R}, b \in B$. Additionally, the set of all connections between flights contained in $\hat{N}$ is given by $\hat{C} = \{(i,j) \in C | i \in \hat{N} \wedge j \in \hat{N}\}$. Replacing $N$ and $C$ with $\hat{N}$ and $\hat{C}$ greatly reduces the time required to solve the column generation subproblem in each iteration.

Branch-and-price is used to solve the RAP to integer optimality. The resulting column generation master problem solution minimises the number of maintenance misalignments given a fixed set of LOFs for aircraft $r \in R \backslash \hat{R}$. Since LOFs are generated only for a subset of aircraft, it is likely that the solution to the RAP is not optimal for the DTAP. As such, the set $\hat{R}$ must be updated following an evaluation of current best solution to the DTAP.

### 3.2.3 Update the variables of the DTAP

The day-one LOFs identified by solving the RAP will reduce the maintenance misalignments in the solution of the DTAP. All columns generated in the RAP are added to the DTAP to re-evaluate the number of maintenance misalignments. Since the column generation master problem is identical to the DTAP, the optimal solution from the RAP is provided to the MIP solver prior to resolving the DTAP. The solution to the DTAP is then used to identify the additional aircraft and flights to add to $\hat{R}$ and $\hat{N}$ respectively.

### 3.2.4 Algorithm termination

The algorithm terminates when no further improvement in the maintenance misalignments can be achieved by adding variables to the DTAP. This is identified using various stopping criteria. First, if the number of aircraft contained in $\hat{R}$ equals the total number of aircraft, then the algorithm is terminated. Second, an *updated* flag, which is *true* only if between iterations the DTAP objective function value decreases or the number of aircraft contained in $\hat{R}$ increases, is used to identify whether the algorithm has stalled. The algorithm terminates if *updated* is *false* for two consecutive iterations. If no columns are added while solving the RAP then the intersection window is increased, the set $\hat{R}$ is updated and the RAP is resolved. This situation is treated as another iteration of the algorithm and if *updated* is *false* the failure count increases and the algorithm will terminate when this count is equal to three.

## 4  Model data

The data required to define the DTAP has been collected from previously performed research, estimates from literature or generated for this study. In regards to the data generated for this study, repeated random experiments are conducted to give a broad overview of the model.

|                                    | F267-A49 | F1165-A289 | F3370-A526 |
|------------------------------------|----------|------------|------------|
| Flights                            | 267      | 1165       | 3370       |
| Aircraft                           | 49       | 289        | 526        |
| Overnight bases                    | 12       | 67         | 73         |
| Maintenance bases                  | 1        | 5          | 10         |
| Initial LOFs                       | 49       | 289        | 526        |
| Total flight times (min)           | 32 040   | 173 833    | 394 155    |
| Avg flight time per aircraft (min) | 653.88   | 601.5      | 749.34     |
| Avg flights per aircraft           | 5.4      | 4.0        | 6.4        |
| Length of day (min)                | 1 440    | 1 440      | 1 080      |

Table 1: The single day flight schedules used as input for the DTAP.

Three single-day flight schedules of vastly different sizes–to be operated by one fleet type—are used for the computational experiments. The details describing the size of the flight schedules, including the number of aircraft, overnight bases and maintenance bases, are presented in Table 1. The flight schedules used in the current experiments have been collected from the study of Maher et al. [18]. In Table 1 the number flights are those operated on a single day by the airline. It is assumed that each airline is operating a cyclic schedule. As such, the input LOFs are said to be repeated on each day of interest. It is trivial to relax this assumption with little change to the difficulty of the problem. Three difficulty measures are provided in Table 1: the total flight times, the average flight time per aircraft and the average number of flights per aircraft. For all flight schedules, each aircraft must perform at least 10 hours of flying during a single day on average. The largest average flight times per aircraft occurs for the F3370-A526 schedule—almost 13 hours per day for each aircraft. For the F3370-A526 flight schedule, the aircraft must also perform 6.4 flights per day, which involves 5.4 turns per day. Assuming that the aircraft turn times are 30 minutes, this means that an aircraft will be busy for $13 + 5.4 \times 0.5 = 15.7$ hours in the day. Given the day length of 18 hours for the F3370-A526 schedule, it is expected that finding a feasible solution to the DTAP will be difficult. Using these metrics, it is believed that the F3370-A526 schedule, which is also the largest schedule, will yield a DTAP that is most difficult to solve. In fact, the difficulty in solving an aircraft routing problem using the F3370-A526 flight schedule is observed in the computational experiments of Maher et al. [18]. Specifically, the solution to the SDAMRP for the F3370-A526 schedule required significantly more runtime than the F267-A49 and F1165-A289 schedules.

The LOFs provided as input to the DTAP are constructed using the one-day routes approach and a classical aircraft routing problem. Both sets of LOFs are collected from the study by Maher et al. [18]. Two different models for generating one-day routes are presented by Maher et al. [18], original (SDAMRP) and recoverable robust formulations (SDAMRP-RR). Only the routes generated from the SDAMRP are used for the current computational experiments. The classical aircraft routing problem used to generate LOFs in Maher et al. [18] is given by a simple modification to the SDAMRP: eliminating the maintenance misalignment penalty term from the objective function. The output from solving the

models presented in Maher et al. [18] is exactly one LOF for each aircraft used to operate the flight schedule. This is indicated in Table 1 where the initial number of LOFs is equal to the number of aircraft.

In the performed experiments the maintenance plan is randomly generated: Assigning each aircraft to receive maintenance exactly once in a six-day period. The six-day maintenance period has been adopted from the work of Heinhold [13] and Maher et al. [18]. The work of Heinhold [13] introduced the one-day routes approach and Maher et al. [18] developed an exact solution approach. Since the input LOFs have been collected from the work of Maher et al. [18] the six-day maintenance period has been used for this study.

To perform an extensive review of the DTAP, 100 different randomly generated maintenance plans have been used as input. A maintenance plan for the DTAP is a six-day schedule that specifies the day each aircraft requires maintenance. Such a schedule can be generated by randomly assigning aircraft to one of the next six days for a maintenance check. To achieve this, the C++ standard `rand()` function is called to generate a sequence of random integers. Starting from day one, an individual aircraft is selected by computing a random number $r$ between 0 and $numAircraft - 1$, where $numAircraft$ is the number of available aircraft. The selected aircraft $r$ is assigned to receive maintenance on the current day of focus. In the case that aircraft $r$ is already assigned to a maintenance check, the next random integer $r$ is selected. When $\lceil numAircraft/6 \rceil$ aircraft are assigned to day one for maintenance, then the process continues for day two. The process terminates when all aircraft are assigned a day to receive maintenance.

It is important to note that the random assignment of maintenance checks is not the method of scheduling maintenance in practice. Maintenance scheduling depends on the work performed by each aircraft prior to solving the DTAP. The random maintenance scheduling in the presented experiments is used to produce a wide range of initial conditions for the DTAP.

The maintenance misalignments are penalised in objective (15) using the cost parameters $c_1^r = 10\,000 \; \forall b \in B, \forall r \in R^b$, $c_2^b = 7000 \; \forall b \in B$ and $c_3^b = 4000 \; \forall b \in B$. The cost parameters are set to weight the importance of the individual maintenance checks. Since an expected maintenance misalignment on day-one has an immediate impact, the penalty parameter is set much higher than the penalty for maintenance misalignments on days two and three. The proposed penalty weights are designed to avoid trade-off between misalignments on day-one and those on subsequent days. In practice, the penalty parameters will be set to suit the preferences of the airline.

Another important objective for an airline is to minimise fuel costs. This is driven by fuel incurring a large proportion of all operational costs. For the presented computational experiments, the input flight schedules are operated by a single fleet type. Making the assumption that aircraft of the same fleet have a similar fuel consumption, all tail assignment solutions are near equal with respect to fuel costs. As such, the cost of fuel is not considered in the current experiments. A point of future research is to

investigate the impact of relaxing this assumption of equal fuel consumption when solving the DTAP.

# 5   Computational experiments

A computational study has been performed to demonstrate the operational performance of the one-day routes business model. In addition, the conducted experiments assess the feasibility of the DTAP with respect to maintenance requirements using fixed inputs and route adjustment approaches. Section 5.1 presents experiments using different fixed LOF inputs and maintenance schedules. Section 5.2 demonstrates the reduction in the maintenance misalignments achieved using the exact and heuristic algorithms for the route adjustment process. Since the iterative algorithm is a novel development of this paper, the performance of this algorithm will guide the discussion in this section. Comparisons and evaluations of the number of maintenance misalignments are made between the standard DTAP formulation, iterative algorithm and branch-and-price algorithm. The evaluation of the route adjustment process is performed with a comparison between the branch-and-price and iterative algorithms in regards to run time and the impact on the input LOFs.

The experiments are performed using the SCIP Optimisation Suite 3.2.0, which includes SCIP 3.2.0 and SoPlex 2.2.0 [9]. The computing infrastructure used for the experiments consists of a cluster of Intel Xeon X5672 CPUs with 3.20 GHz and 48 GB RAM, running Ubuntu 14.04. Each experiment was performed on a single thread exclusively on one node.

## 5.1   Analysing the improved maintenance planning

Experiments are conducted on the DTAP to assess the number of maintenance misalignments when using the two different LOF inputs described in Section 4. The results of Maher et al. [18] suggest that the use of the aircraft routing LOFs should cause maintenance misalignments in the DTAP solution. In contrast, the SDAMRP LOFs are constructed by Maher et al. [18] to significantly reduce—or completely eliminate—the maintenance misalignments.

The results presented in Figure 2 demonstrate a decrease in the number of maintenance misalignments as a result of using the SDAMRP LOFs compared to the aircraft routing LOFs. Surprisingly, in practice many maintenance misalignments still exist when using the SDAMRP LOFs. This is shown by the light grey bars in Figure 2 in the columns representing at least one misaligned aircraft. This highlights the limitation of using fixed LOFs and the need for the development of a route adjustment process.

A summary of the number of maintenance misalignments on days one, two and three after solving the DTAP is presented in Figure 3. These results aim to demonstrate the influence of key features of the DTAP—the day-two and day-three look-ahead maintenance constraints—on maintenance misalignments. The first column of Figure 3 represents the solution to a model that only minimises day-one maintenance misalignments. The second presents the impact of penalising day-two maintenance mis-
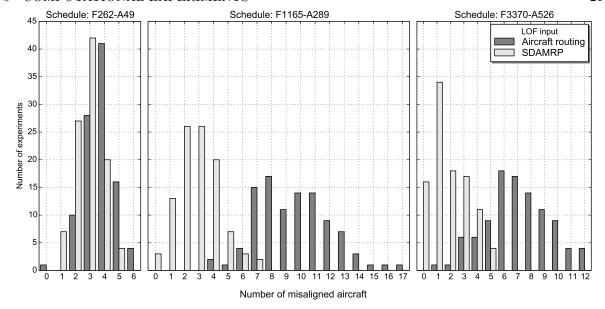
Figure 2: Histogram presenting the number of maintenance misaligned aircraft at the end of day-one over a set of 100 experiments solving the DTAP with fixed LOFs.

alignments. Finally, the third presents the results from solving the DTAP by additionally penalising day-three maintenance misalignments.

A striking observation from Figure 3 is the little interaction between the different features of the DTAP. The increase in the penalty values only impacts the feature directly affected. For example, it is only possible to reduce the day-two or day-three maintenance misalignments through the direct consideration with look-ahead constraints. As such, solving the DTAP to only minimise day-one maintenance misalignments will result in infeasibilities of the maintenance plan on subsequent days. This result suggests that the developed DTAP is valuable for reducing the number of day-one, day-two and day-three maintenance misalignments.

The selection of LOF input is observed to be critical across all metrics presented in Figure 3. The SDAMRP LOFs outperform the aircraft routing LOFs in all metrics except the day-three maintenance misalignments in Case 1 for the F3370-A526 schedule and Case 3 for the F267-A49 schedule. Importantly, the number of day-one maintenance misalignments is significantly reduced using the SDAMRP LOFs compared to the aircraft routing LOFs. Solving the DTAP with the SDAMRP LOFs achieves a reduction in the average number of day-one maintenance misalignments of 22.85%, 69.79% and 73.94% for the F267-A49, F1165-A289 and F3370-A526 flight schedules respectively.

Figure 3 presents a large decrease in the number of day-two and day-three maintenance misalignments when solving the DTAP using the standard settings (Case 3). The average number of day-three maintenance misalignments using the aircraft routing and SDAMRP LOF input decreases from (9, 9), (49, 49), (88, 88) in Case 1 to (1.25, 1.45), (0.54, 0) and (0, 0) in Case 3 for the F267-A49, F1165-A289 and F3370-A526 flight schedules respectively. Similar results are observed for the day-two maintenance
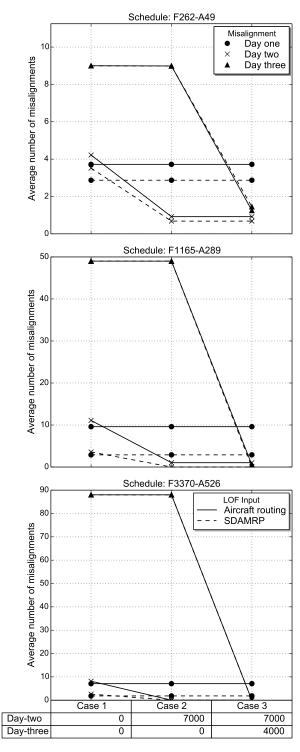
Figure 3: Average maintenance misalignments on days one, two and three using various penalty settings to solve the DTAP.

misalignments. However, the decrease is not as significant. These results suggest the need to consider subsequent days maintenance requirements when using one-day routes as input for the tail assignment.

## 5.2 Evaluating the route adjustment process

The performance of the route adjustment process is assessed with comparisons between the DTAP, branch-and-price and iterative algorithms. A comparison of the day-one maintenance misalignments resulting from the use of each algorithm is presented in Section 5.2.1. The computational performance of the solution algorithms is assessed in regards to the solution run time. The run time comparison is presented in Section 5.2.2. Finally, a comparison of the changes made to the input LOFs when using the iterative algorithm and branch-and-price is presented in Section 5.2.3.

### 5.2.1 Maintenance misalignments

The comparison of the iterative algorithm and the DTAP in regards to the day-one maintenance misalignments is presented in Figures 4 and 5. These figures show a significant decrease in the number of day-one maintenance misalignments by using the iterative algorithm. This effect is particularly evident for the F1165-A289 flight schedule using the aircraft routing LOF input. Solving the DTAP using the F1165-A289 flight schedule results in four to seventeen day-one maintenance misalignments. These misalignments are completely eliminated in all experiments when the iterative algorithm is employed. While this result is also observed when using the SDAMRP LOF input, the decrease in day-one maintenance misalignments is not as great. This is a consequence of the DTAP solution using SDAMRP LOF input exhibiting less day-one maintenance misalignments compared to the aircraft routing LOF input for all flight schedules.

The success of the iterative algorithm is evident in the maximum number of day-one maintenance misalignments across the 100 experiments. For the aircraft routing LOF input the iterative algorithm achieves a maximum of 2, 0 and 2 maintenance misalignments for the F267-A49, F1165-A289 and
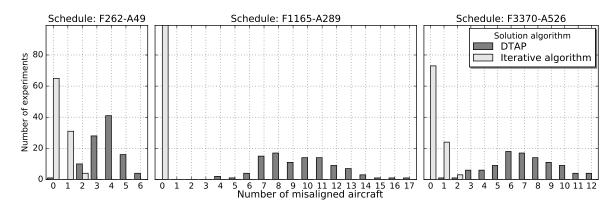


Figure 4: Comparing the DTAP and iterative algorithm with histograms of the number of maintenance misaligned aircraft at the end of day-one over a set of 100 experiments using the aircraft routing LOFs.
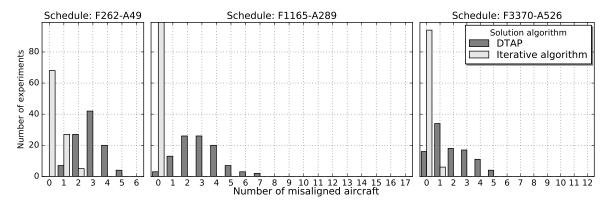
Figure 5: Comparing the DTAP and iterative algorithm with histograms of the number of maintenance misaligned aircraft at the end of day-one over a set of 100 experiments using the SDAMRP LOFs.

F3370-A526 flight schedules respectively. Similarly, for the SDAMRP LOF input the maximum number of maintenance misalignments is 2, 0 and 1 respectively. While the maintenance misalignments are not completely eliminated in all experiments, the decrease is practically significant. Performing one or two over-the-day swaps is much simpler for the airline than rerouting the large number of misaligned maintenance critical aircraft given by the DTAP solution.

Similar to the preceding discussion, Tables 2 and 3 present a comparison of the day-one maintenance misalignments when using the iterative algorithm and a branch-and-price algorithm. The first observation from Tables 2 and 3 is that employing branch-and-price for the F267-A49 flight schedule results in less maintenance misalignments compared to when the iterative algorithm is used. This result demonstrates that the exact algorithm outperforms the heuristic on small instances. Since small instances are typically easy to solve, a heuristic is not required to find good solutions. While the branch-and-price algorithm achieves a better average day-one maintenance misalignment result, the small number of misalignments achieved by both algorithms is acceptable for the tail assignment application.

Contrary to the results for the F267-A49 flight schedule, there is little difference between the performance of the iterative and branch-and-price algorithms for the F1165-A289 and F3370-A526 flight schedules in regards to day-one maintenance misalignments. All instances for the F1165-A289 and F3370-A526 flight schedules are solved within the maximum run time of 7200 seconds for both algorithms. The reduction of all maintenance misalignments for the F1165-A289 flight schedule is a positive

| Aircraft routing LOFs | F267-A49 | | F1165-A289 | | F3370-A526 | |
| Misalignments | BnP | Iter. | BnP | Iter. | BnP | Iter. |
|---|---|---|---|---|---|---|
| 0 | 88 | 65 | 100 | 100 | 76 | 73 |
| 1 | 12 | 31 | 0 | 0 | 21 | 24 |
| 2 | 0 | 4 | 0 | 0 | 3 | 3 |
| $\geq 3$ | 0 | 0 | 0 | 0 | 0 | 0 |

Table 2: Number of experiments exhibiting a given level of maintenance misalignments when using the branch-and-price and iterative algorithms with the aircraft routing LOFs.

| SDAMRP LOFs | F267-A49 | | F1165-A289 | | F3370-A526 | |
|:---:|:---:|:---:|:---:|:---:|:---:|:---:|
| Misalignments | BnP | Iter. | BnP | Iter. | BnP | Iter. |
| 0 | 100 | 68 | 100 | 100 | 95 | 94 |
| 1 | 0 | 27 | 0 | 0 | 5 | 6 |
| 2 | 0 | 5 | 0 | 0 | 0 | 0 |
| ≥ 3 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 3: Number of experiments exhibiting a given level of maintenance misalignments when using the branch-and-price and iterative algorithms with the SDAMRP LOFs.

result for both algorithms. This demonstrates that optimal maintenance planning solutions exist for a wide range of maintenance schedules. The results presented for the F3370-A526 flight schedule show the difficulty in satisfying the maintenance requirements for all aircraft. However, since a the reduction in maintenance misalignments is achieved, the effort required on the day of operations to perform further aircraft swaps is reduced. The results presented in Tables 2 and 3 indicate that the iterative algorithm is a very competitive heuristic for the route adjustment DTAP compared to the exact branch-and-price algorithm.

### 5.2.2 Run time comparison

The run times for the 100 experiments with the DTAP, iterative and branch-and-price algorithms using the aircraft routing and SDAMRP LOF inputs are presented in Figure 6. An important observation from Figure 6 is the very short run times for each of the algorithms across the majority of experiments. All experiments for the DTAP algorithm using the F267-A49, F1165-A289 and F3370-A526 flight schedule terminate in a maximum of, respectively, 0.2, 1.44 and 76.78 seconds for the aircraft routing LOF input and 0.14, 5.54 and 76.8 seconds for the SDAMRP LOF input. Comparatively, the iterative algorithm requires much longer run times but still very small in magnitude. The iterative algorithm requires at most 1.91, 53.34 and 133.31 seconds for the F267-A49, F1165-A289 and F3370-A526 flight schedules respectively. While this represents a significant increase in run times, it is acceptable given the observed decrease in the number of maintenance misalignments achieved by the iterative algorithm as presented in Figures 4 and 5.

The benefit of the iterative algorithm is evident when comparing the run times with that of the branch-and-price algorithm. Figure 6 shows the iterative algorithm achieves a smaller run time on average compared to branch-and-price for the F267-A49 and F1165-A289 flight schedules. The same benefit is not observed for the F3370-A526 flight schedule where the branch-and-price approach achieves a better run time performance for some instances. The iterative algorithm achieves a better average run time when the aircraft routing flight schedule is used, specifically 42.67 seconds compared to 50.44 seconds. Conversely, the average run time when the SDAMRP input is used is worse for the iterative algorithm, 30.94 seconds compared to 28.70 seconds. Interestingly, the maximum run time for the
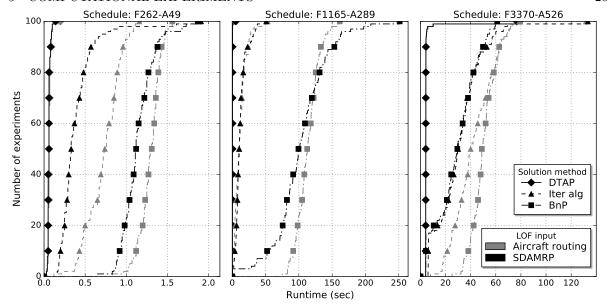
Figure 6: The run time required to solve 100 tail assignment instances using the DTAP, iterative algorithm and branch-and-price algorithm.

iterative algorithm is significantly greater than that for the branch-and-price algorithm using both inputs for the F3370-A526 flight schedule. This is due to a single instance where the branch-and-price algorithm solves the tail assignment problem at the root node. For this particular instance the branch-and-price algorithm solves in a run time less than that required for the DTAP. As such, it is not possible for the iterative algorithm to achieve any improvement on this instance. Overall, the maintenance misalignment reduction presented in Section 5.2.1 and run time comparison given by Figure 6 demonstrates that the iterative algorithm is a very useful practical approach for reducing the number of day-one maintenance misalignments when compared with the branch-and-price algorithm.

The initial solution given by the input LOFs provides a significant algorithmic benefit. In comparison to the runtimes of the SDAMRP presented in Maher et al. [18], the solution times for the DTAP are orders of magnitude less than that for the SDAMRP. Specifically, the runtime to solve the SDAMRP for the F3370-A526 schedule is reported as 65,087 seconds [18] compared to the maximum runtime of 61.13 seconds for solving the DTAP by branch-and-price using the SDAMRP LOF input. While the runtime difference is largest for the F3370-A526 schedule, the benefit from using the LOF input for the DTAP is also observed for both the F267-A49 and F1165-A289 schedules.

The performance improvements provided by the input LOFs explain the small difference in the runtimes between the branch-and-price and iterative algorithms. Since the branch-and-price algorithm exhibits good runtimes, there is limited gains that can be made through a heuristic approach. However, the benefits of such a heuristic approach is still observed in the results generated using the F1165-A289 schedule. These benefits may suggest that in cases where the input LOFs are far from suitable for satisfying maintenance requirements, the iterative algorithm can significantly reduce the solution

runtimes.

### 5.2.3   Impact on input LOFs

A comparison between the exact branch-and-price algorithm and the iterative algorithm is presented in
Tables 4 and 5. The metrics for this comparison are i) the number of connection changes between the
input LOFs and the final tail assignment solution and ii) the number of columns added. The results
are presented as the average over 100 experiments with random maintenance plans for both the aircraft
routing and SDAMRP LOF input.

   The results presented in Tables 4 and 5 demonstrate that the iterative algorithm is very competitive
with branch-and-price in both metrics. Most importantly, pairing these results with those presented in
Sections 5.2.1 and 5.2.2 shows that the iterative algorithm reduces the maintenance misalignment with
reduced computational effort. As such, this algorithm has demonstrable value for solving the DTAP and
reducing maintenance misalignments.

   Only a small number of connection changes are reported for both the branch-and-price and iterative
algorithms in Tables 4 and 5. This is important from the perspective of minimising the impact that
modifying LOFs has on other airline resources that rely on the aircraft routing solution, such as crew.
The number of connection changes that are performed as part of the route adjustment process is very
small compared to the total number of used connections. In the worst case, which is for the F267-A49
schedule using the aircraft routing LOFs, the percentage of connection changes is 7.62%. This is due
to the limited number of maintenance locations and hence more rerouting is required to correct for
any schedule perturbations. In the best case—F3370-A526 schedule using the SDAMRP LOFs—the
percentage of connection changes is 0.24%. This is a negligible change to the aircraft routing solution.
Thus, it is expected that little change to the associated crew pairing solution will be required.

   The small number of connection changes points to the value of the one-day routes business model.
By only planning for a single day, only small changes are required to adjust the aircraft routing solution
to ensure all maintenance requirements are satisfied. An important result is the significant decrease
in the number of connection changes for the F1165-A289 schedule when using the SDAMRP LOFs.
This provides a clear indication that using one-day routes in planning helps to reduce the number of

| Aircraft routing LOFs | | F267-A49 | F1165-A289 | F3370-A526 |
|---|---|---|---|---|
| Connection Changes | Branch-and-price | 16.62 (7.62%) | 22.4 (2.56%) | 21.5 (0.76%) |
| | Iterative algorithm | 16.52 (7.58%) | 20.76 (2.37%) | 22 (0.77%) |
| LOFs Added | Branch-and-price | 1165.9 | 7077.28 | 4013.94 |
| | Iterative algorithm | 512.56 | 1857.68 | 2070.58 |

Table 4: The average number of connection changes (percentage of total used connections) and columns
added over 100 experiments when solving the DTAP with branch-and-price and the iterative algorithm
using the aircraft routing LOFs.

| SDAMRP LOFs | | F267-A49 | F1165-A289 | F3370-A526 |
|---|---|---|---|---|
| Connection Changes | Branch-and-price | 14.8 (6.79%) | 6.74 (0.77%) | 6.74 (0.24%) |
| | Iterative algorithm | 14.68 (6.73%) | 6.8 (0.78%) | 6.78 (0.24%) |
| LOFs Added | Branch-and-price | 1352.2 | 7442.89 | 2407.53 |
| | Iterative algorithm | 457.36 | 1946.17 | 1343.87 |

Table 5: The average number of connection changes (percentage of total used connections) and columns added over 100 experiments when solving the DTAP with branch-and-price and the iterative algorithm using the SDAMRP LOFs.

maintenance misalignments and the required changes as a result of schedule perturbations.

An advantage of the iterative algorithm is its application in integrated settings. The route adjustment problem is solved independently of the original problem, as such it is possible to apply this solution approach when integrating the DTAP with an additional planning problem that uses LOFs as input. For example, the gate assignment problem is solved using the DTAP solution as input. As such, these two problems could be integrated and the route adjustment process applied. Since the route adjustment process is solved independent of the target problem, the identified LOFs can be used as input for both the gate assignment problem and the DTAP. In such an integrated setting, the development of an effective solution algorithm for the integrated problem requires an efficient route adjustment problem.

In regards to the efficiency of the route adjustment problem an important comparison metric for the iterative algorithm is the number of columns added. While the exact branch-and-price algorithm is shown in Section 5.2.2 to require only small run times for most experiments, this may not be the case when applying the DTAP in integrated settings. Since the iterative algorithm generates significantly less columns, improved efficiency is expected when integrating the DTAP with other airline planning processes.

# 6   Conclusions

The DTAP is an important and necessary stage in the airline planning process. The business model of using one-day routes for short term maintenance planning is the focus of this paper. In particular, an operational optimisation problem tasked with the overnight assignment of aircraft to LOFs is developed. The presented DTAP aims to minimise the number of maintenance misalignments on days one, two and three. To further reduce the number of maintenance misalignments with little impact to the input LOFs, an exact branch-and-price algorithm and a heuristic iterative algorithm to reconstruct LOFs are presented. These algorithms both employ branch-and-price with the iterative algorithm only generating aircraft routes for a subset of the considered fleet. The results demonstrate the effectiveness of the one-day routes business model through the ability of the DTAP and route adjustment process to reduce the number of maintenance misalignments for a given set of LOFs.

The DTAP is formulated with constraints that implicitly satisfy maintenance requirements on days two and three. This implicit method is shown to be an effective method to plan maintenance with a model that is solvable by state-of-the-art mixed integer programming solvers. Further, this modelling approach can be generalised to maintenance planning problems that span across multiple time periods. It is demonstrated that solving the DTAP without look-ahead constraints results in a large number of maintenance misalignments across a three day period. Additionally, even with the look-ahead constraints the maintenance misalignments resulting from solving the DTAP indicate the need to develop a route adjustment process. The exact and heuristic algorithms developed are shown to be very effective in reducing the number of maintenance misalignments. The results demonstrate that the iterative algorithm is very competitive compared to the branch-and-price algorithm. The strength of the iterative algorithm is the use of mixed-integer programming solvers to guide the search for improving aircraft routes.

The integration of tasks within the airline planning process is a critical development for high quality solution approaches. The presented work integrates part of the planning process, the overnight aircraft assignment, with an operational problem of reoptimising aircraft routes. Future research involves identifying tasks for integration with the DTAP. Such tasks include achieving equal utilisation of the fleet and managing the non-uniform costs of aircraft of various ages. Also, the iterative algorithm presents a practical solution approach to aid the integration of maintenance planning with other operational problems. The investigation into the use of the iterative algorithm in an integrated setting is a focus of future work.

# Acknowledgements

# References

[1] Y. Ageeva. Approaches to incorporating robustness into airline scheduling. Master's thesis, Massachusetts Institute of Technology, August 2000.

[2] M. Başdere and Ümit Bilge. Operational aircraft maintenance routing problem with remaining time consideration. *European Journal of Operational Research*, 235(1):315–328, 2014.

[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] R. Borndörfer, I. Dovica, I. Nowak, and T. Schickinger. Robust tail assignment. In *Proceedings of the fiftieth annual symposium of AGIFORS*, 2010.

[5] L. Clarke, E. Johnson, G. Nemhauser, and Z. Zhu. The aircraft rotation problem. *Annals of Operations Research*, 69:33–46, 1997.

[6] 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.

[7] T. Feo and J. Bard. Flight scheduling and maintenance base planning. *Management Science*, 35(12):1415–1432, 1989.

[8] G. Froyland, S. J. Maher, and C.-L. Wu. The recoverable robust tail assignment problem. *Transportation Science*, 48(3):351–372, 2014.

[9] G. Gamrath, T. Fischer, T. Gally, A. M. Gleixner, G. Hendel, T. Koch, S. J. Maher, M. Miltenberger, B. Müller, M. E. Pfetsch, C. Puchert, D. Rehfeldt, S. Schenker, F. Serrano, Y. Shinano, S. Vigerske, D. Weninger, M. Winkler, J. T. Witt, and J. Witzig. The SCIP optimization suite 3.2. Technical Report ZIB Report 15-60, Zuse Institute Berlin, 2016.

[10] R. Gopalan and K. T. Talluri. The aircraft maintenance routing problem. *Operations Research*, 46(2):260–271, 1998.

[11] M. Grönkvist. *The Tail Assignment Problem*. PhD thesis, Chalmers University of Technology and Göteborg University, August 2005.

[12] M. Haouari, S. Shao, and H. D. Sherali. A lifted compact formulation for the daily aircraft maintenance routing problem. *Transportation Science*, 47(4):508–525, 2013.

[13] A. Heinhold. Maintenance reachability at Southwest Airlines. In *Proceedings of the AGIFORS Scheduling and Strategic Planning Study Group Meeting, Dallas*, 2008.

[14] E. Lacasse-Guay, G. Desaulniers, and F. Soumis. Aircraft routing under different business processes. *Journal of Air Transport Management*, 16(5):258–263, 2010.

[15] M. Lapp and A. Cohn. Modifying lines-of-flight in the planning process for improved maintenance robustness. *Computers & Operations Research*, 39(9):2051–2062, 2012.

[16] Z. Liang and W. A. Chaovalitwongse. A network-based model for the integrated weekly aircraft maintenance routing and fleet assignment problem. *Transportation Science*, 47(4):493–507, 2013.

[17] Z. Liang, Y. Feng, X. Zhang, T. Wu, and W. A. Chaovalitwongse. Robust weekly aircraft maintenance routing problem and the extension to the tail assignment problem. *Transportation Research Part B: Methodological*, 78:238–259, 2015.

[18] S. J. Maher, G. Desaulniers, and F. Soumis. Recoverable robust single day aircraft maintenance routing problem. *Computers & Operations Research*, 51:130–145, 2014.

[19] S. Ruther. *Integrated aircraft routing, crew pairing, and tail assignment.* PhD thesis, The University of Newcastle, March 2013.

[20] D. M. Ryan and B. A. Foster. An integer programming approach to scheduling. In A. Wren, editor, *Computer Scheduling of Public Transport: Urban Passenger Vehicle and Crew Scheduling*, pages 269–280. North-Holland Publishing Company, Amsterdam, 1981.

[21] C. Sriram and A. Haghani. An optimization model for aircraft maintenance scheduling and re-assignment. *Transportation Research Part A: Policy and Practice*, 37(1):29–48, 2003.

[22] K. Talluri. The four-day aircraft maintenance routing problem. *Transportation Science*, 32(1):43–53, 1998.

[23] O. Weide, D. Ryan, and M. Ehrgott. An iterative approach to robust and integrated aircraft routing and crew scheduling. *Computers & Operations Research*, 37(5):833–844, 2010.