Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 00 (2026) 000–000

Transportation Research Procedia

www.elsevier.com/locate/procedia

Euro Working Group on Transportation Annual Meeting 2025 - EWGT2025

Flexible maintenance in aircraft rotation models: a modular component of the digital airline twin

Sarah Albrecht[a,*], Milena Röhrs[a], Thorsten Ehlers[a], Klaus Lütjens[a]

[a] *Institute of Air Transport, German Aerospace Center (DLR), 21079 Hamburg, Germany*

Abstract

The planning processes of an airline are complicated due to the multitude of options. Additionally, it takes a long period of time from the first planning steps, like creating the network and defining the fleet, until the flights are executed. The development of the digital airline twin (DAT) at the German Aerospace Center (DLR) is motivated by the objective of simulating and optimizing these processes. The DAT will be a general tool which can be used for airlines with different business strategies and to answer various research questions. By dividing the processes into smaller sub-processes, each model can calculate larger instances and feedback loops can be included to iterate the results. One of these planning processes is the generation of aircraft rotations, which are sequences of specific flights within a given time frame that are later assigned to individual aircraft, known as aircraft rotation problem. This includes the necessity for sufficient availability for regular maintenance for each rotation in order to ensure operational efficiency and safety. To enable flexible, feedback-driven use within the DAT, different simplifications are implemented that leave space for further, more detailed airline planning steps. This paper presents two approaches. Both including sufficient availability for maintenance in each aircraft rotation for an airline to later schedule specific maintenance events. Moreover, the results regarding a one-week flight schedule are compared and the suitability of the approaches for the DAT is discussed.

Keywords: aircraft rotation; digital twin; optimization; maintenance.

---

1. Introduction

Due to the large number of flights and airports as well as different types of aircraft, organizing the processes of an airline is a complex task. On the one hand, the long-term steps of planning, such as creating the flight network and determining the fleet, begin months or even years before departure. On the other hand, there are short-term changes on the day of operation of the flight due to adverse weather conditions or other unplanned changes. These different

---

[*] Corresponding author. Tel.: +49-40-2489641-224
E-mail address: sarah.albrecht@dlr.de

time periods are one reason why the field of airline planning processes has been relevant for decades and still is, as discussed by Mirjafari et al. (2023).

In order to optimize these planning processes, it is necessary to understand and model every step of them. This is the aim of the digital airline twin (DAT) which is developed at the German Aerospace Center (DLR). These processes include e.g. the fleet planning and assignment, the aircraft rotation and the crew scheduling. The DAT will model the different planning processes either individually or in an integrated manner. In the end, the user should be able to decide which module to use for the calculation. This way, it will be possible to use one module to execute a specific planning process or combinations of modules to answer various research questions. Additionally, the aim of the DAT is to check the feasibility of a result from one planning step as an input for the next one. This includes the possibility to give feedback to the previous tool if the given solution is not compatible. According to the feedback, the previous tool might be executed with different assumptions or additional constraints to address the infeasibility in the next step.

This paper focuses on modeling the aircraft rotation problem in the context of the DAT. This means defining sequences of specific flights (routes) based on a flight schedule that can be flown by one aircraft (Fig. 1). A relevant criterion for feasible routes is the inclusion of sufficient availability for maintenance events for later, more detailed planning steps. The implementation in the DAT requires simplified models that are realistic enough to build further planning steps on them but do not make extensive demands on the inputs, as these should come from interchangeable models. Another important factor is the runtime, as the model can be part of feedback loops and therefore has a significant influence on the overall runtime of the DAT. This paper presents two approaches to schedule enough availability for maintenance and compare these regarding their suitability for the DAT. In contrast to the existing models, the maintenance events are not scheduled exactly but the time each aircraft will be available at a maintenance station is considered. This gives the opportunity to plan more detailed sub-processes and include feedback loops.

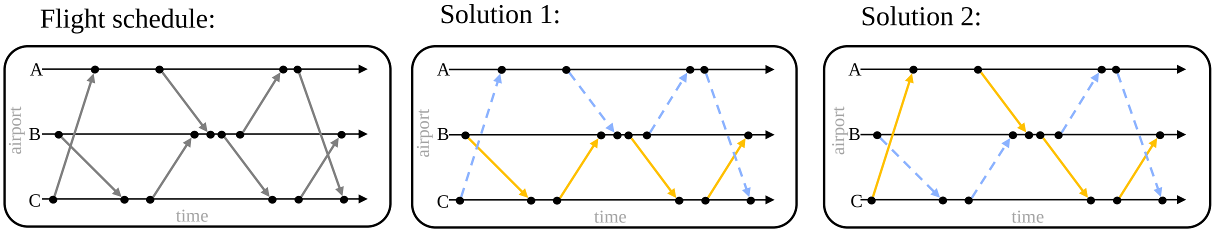


Fig. 1. Two possible rotations with two aircraft (blue/dashed and yellow/solid) to a given flight schedule (gray) with three airports A, B and C.

1.1. Literature review

In the last years there have been different approaches generating aircraft rotations including variations of maintenance constraints. For example Clarke et al. (1997) modeled the aircraft rotation problem as an Euler tour through an Eulerian digraph. The model is solved using Lagrangian relaxations. Ruan et al. (2021) implemented a method which uses reinforcement learning to generate routes in a connection network. The model considers the flight time and number of take-offs between two maintenance checks and the capacity at the maintenance stations as maintenance constraints. Another approach using a connection network is presented in Safaei and Jardine (2018) and considers the maintenance as individual tasks. These are validated through a sliding window which checks the capacity and the required working hours. In the model, it is possible to add flights for reaching the maintenance airport and a penalty is included if maintenance is not possible. Lagos et al. (2020) considers unit-times per task to limit the different times of resources like facilities or workers. Different heuristic approaches using e.g. simulated annealing, genetic algorithms or a very large-scale neighborhood search heuristic to solve the aircraft maintenance routing problem are presented by Eltoukhy et al. (2017) and Al-Thani et al. (2016). An alternative approach for a cyclic, daily schedule using a time-space network (TSN) is presented in Liang et al. (2011).

Often the aircraft rotation problem is combined with an adjacent planning process. For example, it is combined with fleet assignment for a one-week schedule in Liang and Chaovalitwongse (2013) or with tail assignment in Khaled et al. (2018). According to the current settings at AirFrance, Parmentier and Meunier (2020) formulated the aircraft rotation as an integer problem combined with crew pairing. It requires that each aircraft is maintained once every four days.

In difference to the existing models, the presented model schedules the availability for required maintenance but does not plan specific maintenance events. The detailed planning of the maintenance events and tail assignment is left to the next step to split the planning process into smaller sub-processes. This way the interplay between the processes within the whole airline planning procedure can be understood and the possibility for feedback loops between the sub-processes is given. The existing models often work with one-day or periodic schedules which limits the generality. In contrast, the presented model is tested with an one-week schedule while applicable to arbitrary schedule length. By this kind of simplification, coupled with a feedback loop for validation, a more favorable scaling behavior is expected.

2. Methodology

2.1. Modeling aircraft rotation problem

The schedule is modeled as a TSN by a directed graph, as in Liang et al. (2011). Each node $v \in V$ represents a flight activity such as arrival or departure. These nodes are connected by directed edges $e \in E$ which are categorized as flight arcs, ground arcs or maintenance arcs. Ground arcs connect successive nodes of each airport to keep track of all aircraft on the ground. The maintenance arcs are specific ground arcs which indicate maintenance availability during the night and are defined accordingly to the maintenance criteria used. The flight arcs represent the flight legs $l \in L$ and are defined by the start and end node as well as a key which consists of flight number, origin and destination. Moreover, each flight arc has the assigned fleet type $f \in F$ as an attribute. Additionally, there is a source and sink node which is connected to the first or last node of each airport to achieve consistency without the necessity for a periodic schedule (Fig. 2).

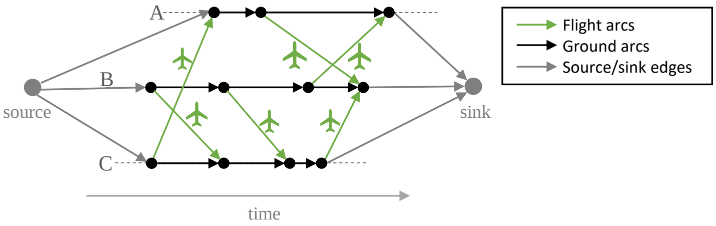


Fig. 2. A TSN with three airports A, B and C.

The optimization problem, which is described in detail in section 2.3, is formulated as a mixed-integer problem and solved using Gurobi (Gurobi Optimization, LLC (2024)). The objective function minimizes the number of needed aircraft. This simultaneously maximizes the utilization per used aircraft because the flight time is divided by as few aircraft as possible. The flight time is defined as the the block time which includes the actual flight time as well as the taxi-in and taxi-out time. Between two flights, there must be at least the minimal turn-around time (TAT) according to the fleet type in order for the route to be feasible. This time is required to prepare the aircraft for the next flight. It includes the boarding and deboarding of passengers, refueling, pre-flight checks of the aircraft and preparation of the aircraft by the new crew.

2.2. Input data

The input of the model is a set of flight legs $l \in L$ of an airline with assigned fleet types $f \in F$. An example for one single input element is ”HAM-LHR, 2023-04-17 07:00, 2023-04-17 08:40, A320”. This declares a flight from

Hamburg (HAM) to London Heathrow (LHR) at April 17th 2023 which starts at 7.00 am, arrives at 8.40 am and is flown by an A320. In contrast to the model by Liang et al. (2011), the schedule does not need to be periodic to achieve more generality within the DAT. Additionally, the model receives the number of aircraft $a \in A$ per assigned fleet type of the airline from Cirium (2025) as an upper bound for the possible number of rotations. Moreover, for each fleet type the minimal TAT is taken from the aircraft manuals (see Airbus (2025), Boeing (2025)).

2.3. Basic constraints

In order to generate physically realizable rotations, some constraints are required in the model which are described in this subsection. Additionally, some maintenance constraints are required according to the chosen maintenance approach. These are described separately in the following subsection 2.4. The model includes some sets which are defined by the input and the consequential TSN. These are the basis for the following variables and constraints.

$$
\begin{aligned}
V' := V \cup \{v_{source}, v_{sink}\}&: \text{Set of all nodes including source and sink}\\
E&: \text{Set of all edges}\\
F&: \text{Set of all fleet types}\\
A&: \text{Set of all aircraft}\\
L&: \text{Set of all flight legs}
\end{aligned}
$$

The assignment variable x_{al} defines whether an aircraft a is assigned to a flight leg l (Eq. 1). The variable y_{ea} assigns an aircraft a to an edge e regardless of the type of the arc (Eq. 2). This variable creates the connection between the TSN and the constraints within the optimization model. Moreover, the input specifies the assigned fleet type f to each leg l which is translated into the parameter z_{fl} to be used within the model (Eq. 3). Accordingly, the mapping of the aircraft a to the fleet type f is stored in parameter η_{af} (Eq. 4).

$$x_{al} = \begin{cases} 1 & \text{if aircraft } a \text{ is assigned to leg } l \\ 0 & \text{otherwise} \end{cases} \quad \forall\, a \in A, l \in L \tag{1}$$

$$y_{ea} = \begin{cases} 1 & \text{if aircraft } a \text{ is assigned to edge } e \\ 0 & \text{otherwise} \end{cases} \quad \forall\, e \in E, a \in A \tag{2}$$

$$z_{fl} = \begin{cases} 1 & \text{if fleet type } f \text{ is assigned to leg } l \\ 0 & \text{otherwise} \end{cases} \quad \forall\, f \in F, l \in L \tag{3}$$

$$\eta_{af} = \begin{cases} 1 & \text{if aircraft } a \text{ is assigned to fleet type } f \\ 0 & \text{otherwise} \end{cases} \quad \forall\, a \in A, f \in F \tag{4}$$

The objective (5a) of the model is to minimize the sum over the activity ξ_a over all aircraft. Their activity is defined as a binary variable for each aircraft (5h). If an aircraft is assigned to at least one leg of the schedule, it is defined as active by constraint (5b). The flight coverage is ensured by constraint (5c) which asserts that exactly one aircraft a is assigned to each flight leg l. An aircraft a can only be assigned to a leg l if it is of the required type f as checked by constraint (5d). The flow balance within the TSN is covered by the constraint (5e). It checks for every node v except the source and sink node that an aircraft a is assigned to the same number of incoming and outgoing edges. The set of incoming edges of a node v is defined as in_v and accordingly all outgoing edges are in out_v. Constraint (5f) ensures that each aircraft is assigned to at most one incoming edge of the sink in_{sink} which assures that each aircraft is only used once. To assert that the time between a departure and an arrival is at least the minimal TAT, constraint (5g) is included in the model. A leg l_1 is defined as smaller than an other leg l_2 if the arrival of l_1 takes place at the same airport earlier than the departure of l_2. For each pair of legs with $l_1 < l_2$ where the departure $l_{1,dep}$ is not later than the arrival $l_{2,arr}$ plus the TAT only one of the flights is possible for the aircraft. This means that at most one of the variables x_{al_1} or x_{al_2} for a specific aircraft a can be set to 1.

$$\min \quad \sum_{a \in A} \xi_a \tag{5a}$$

$$\text{s.t.} \quad \sum_{l \in L} x_{al} \leq \xi_a \cdot |L| \; \forall\, a \in A, \tag{5b}$$

$$\sum_{a \in A} x_{al} = 1 \forall l \in L, \tag{5c}$$

$$\eta_{af} \cdot x_{al} \leq z_{fl} \; \forall\, a \in A, l \in L, f \in F, \tag{5d}$$

$$\sum_{i \in in_v} y_{ia} - \sum_{o \in out_v} y_{oa} = 0 \; \forall\, a \in A, v \in V, \tag{5e}$$

$$\sum_{i \in in_{sink}} y_{ia} \leq 1 \; \forall\, a \in A, \tag{5f}$$

$$x_{al_1} + x_{al_2} \leq 1 \; \forall\, a \in A, l_1, l_2 \in L |\, l_1 < l_2 \wedge l_{1,arr} + TAT_a > l_{2,dep}, \tag{5g}$$

$$\xi_a \in \{0, 1\} \; \forall\, a \in A \tag{5h}$$

2.4. Maintenance constraints

If the rotations of the aircraft are planned without considering maintenance at all, this might lead to infeasibility in the later maintenance planning step. Therefore, some constraints need to be included in the model. The maintenance checks are usually planned to take place at night to not interrupt the flight plan. This requires that in a given time frame each aircraft must spend a minimum specified duration overnight at designated airports equipped with maintenance facilities. These airports are called maintenance stations, given by the set $m \in M$. Within the model, nighttime is defined between 10.00 pm and 6.00 am. The implementation provides two options to enforce the availability for regular maintenance events. Firstly, this is done by ensuring that each aircraft spends a user-defined time at maintenance airports during the nights. Alternatively, the user sets a value of the maximal days between two nights at a maintenance station in the rotations which implies sufficient availability to maintain each aircraft. We use the following sets and parameter for modeling these maintenance constraints:

M: Set of all maintenance stations
$q \in \mathbb{N}$: Number of scheduled days
$N := \{1, ..., q\}$: Set of indices of nights in the schedule

Both of these options require a variable to track the availability at a maintenance airport for each aircraft. The variable w_{amn} is set if the aircraft a is at the maintenance station m during the night n (Eq. 6). The first night is defined as the night before the first flight departs and the last is the night before the last scheduled day. So the number of nights is always the same number as the scheduled days.

$$w_{amn} = \begin{cases} 1 & \text{if aircraft } a \text{ is at maintenance airport } m \text{ at night } n \\ 0 & \text{otherwise} \end{cases} \quad \forall\, a \in A, m \in M, n \in N \tag{6}$$

2.4.1. *First maintenance approach*

The first option sums the hours that each aircraft a spends at a maintenance station m. To do so, all the ground edges during the night time are marked as maintenance arcs. The sum over the duration of all these edges at all maintenance stations for each aircraft must be greater than a predefined lower bound lb. This is specified in constraint (7). The length of each edge e is labeled as $len(e)$ and is defined in hours.

$lb \in \mathbb{R}^+$: Minimal time to be available for maintenance in the schedule interval

$E_m \subset E$: Maintenance arcs

$$\sum_{m \in M} \sum_{e_m \in E_m} y_{e_m a} \cdot len(e_m) \geq lb \; \forall \; a \in A \tag{7}$$

2.4.2. *Second maintenance approach*

The second approach is based on the assumption that there are mostly no overnight flights and that an aircraft stays at the same airport all night. This implies that there is enough time for maintenance checks if an aircraft is on the required station. Therefore, only one edge during night time needs to be marked as maintenance arc and the real time at the airport must not be considered. However, it is required that there are at most δ_{maint} days between two nights at a maintenance station. The value δ_{maint} can be defined by the user.

To check the criteria for every interval, a sliding window of length δ_{maint} is defined in constraint (8). It verifies for every possible position k that each aircraft a is at a maintenance station m over night at least once. The possible positions are length of the network q subtracted by δ_{maint}.

$$\delta_{maint} \leq q, \text{ with } \delta_{maint} \in \mathbb{N}_{>0}$$

$$\sum_{i=k}^{k+\delta_{maint}} \sum_{m \in M} w_{ami} \geq 1 \; \forall \; a \in A, k \in [0, \; q - \delta_{maint}] \tag{8}$$

3. Results

The functionality of the aircraft rotation model is demonstrated through the calculation of a one-week schedule. Particularly, the difference between the results of the two maintenance simplifications is elaborated in the following. Moreover, it is shown that the model generates valid aircraft rotations for both conditions which means that every route fulfills the assumed restrictions and that each aircraft is long enough available for planning maintenance in further steps.

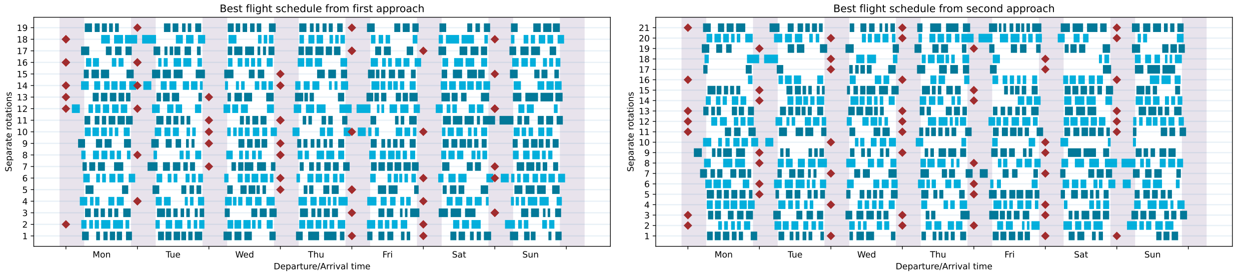
As the aircraft rotation model is part of the DAT, the required input is calculated by the previous model. In this case, the previous planning step is the schedule repair (A2) and fleet assignment presented by Röhrs et al. (2025). Flight legs of a large German airline of the week starting on January 16th 2023 were taken from Sabre GLBL Inc. (2025). In fleet assignment, a fleet type is assigned to each leg based on assignment dependent cost factors. This one-week schedule was filtered by one fleet type in order to create the fleet types sub-schedule. For those sub-schedules, the aircraft rotations can be generated separately as the type-dependent constraints are not interconnected. For the calculations the default settings of Gurobi are used except that the number of threads was limited to eight.

The sub-schedule used includes 659 flight legs. The duration of the flight legs are all less than five hours. The minimal possible number of aircraft to execute this flight schedule without considering maintenance is 16. This includes the flight time and the minimal TAT. For the first approach the lower bound lb is set to 15 hours which corresponds to about two nights. Accordingly, the maximal number of days between two nights at a maintenance station δ_{maint} for the second approach is set to 3 days. This way it is possible to execute the schedule with two nights at a maintenance station in both cases to achieve comparability. The results are presented in Tab. 1.

Using the constraints of the first approach, the model finds two feasible solutions. It was possible for the solver to find an optimal solution. The minimal number of used aircraft is 19 which are 3 more than without maintenance

Table 1. Results of the solver for a one-week schedule with 659 legs.

	Number of solutions	Explored nodes	Run time	Min. needed aircraft	Max. needed aircraft
Approach 1	2	818	17 396s	19 (optimal)	21
Approach 2	9	19 346	> 413 981s	21	25


Best flight schedule from first approach

Best flight schedule from second approach

Fig. 3. Best flight schedule of first approach (left) and second approach (right).

constraints. The maximum amount of time between two nights at a maintenance station is six days meaning that at the latest it is possible to maintain each aircraft every sixth night. The overall time spend at the maintenance stations is distributed among two or three days for each aircraft.

With the second approach, the model finds nine feasible solutions. Even though the solver explores more than 20 times as many nodes as in the first approach, the calculation did not terminated. Due to the minimal number of needed aircraft, the result is expected to be acceptable near to the optimum. The run time is nearly proportional to the number of explored nodes. The closest to optimal sets of rotations out of the found solutions would need 21 aircraft to execute the schedule. The solution furthest from the optimal needs 25 aircraft. Each aircraft is available for maintenance two or three nights within the flight schedule. Due to the defined length of a night, the solutions would be valid for the first approach but vice versa this is not the case.

The best result of each approach, meaning the ones with the minimal number of rotations, is shown in Fig. 3. The model finds rotations for the given fleet assignment in both approaches. Each rotation is represented as one horizontal line with blue bars representing the flights. The night times are shaded in gray. The red diamonds represent each night when the aircraft is available for maintenance. The plot on the left site shows the result of the first approach, whereas the right plot shows the result of the second approach. On the one hand, it is noticeable that the utilization on the left is more homogeneously distributed than on the right site but the maintenance availability is often at successive nights with longer periods in between. For example, the rotation number 16 can only be maintained in the first two nights. This may not fit all requirements for the next, more detailed maintenance planning step. Depending on the network structure of the airline and the amount of flights which are connected to a maintenance station, this may not be an issue in every case. If most flights start or end at a maintenance station, most of the nights are spend at these airports and the regular availability is not a limiting factor. Moreover on the left site, some overnight stays are short due to late arrivals and early departures, making it questionable whether maintenance could actually take place. This can be seen for instance in rotation number 14 during the second night. The last flight arrives at 11.35 pm and the next flight departs at 2.00 am. Subtracting the TAT, less then two hours remain for maintenance in this night. Due to potential taxi time to bring the aircraft to the maintenance facilities, the aircraft is probably not available at all for maintenance. On the other hand, in the second approach some rotations are planned with longer ground times which also causes other problems beside the optimal utilization. One example for these problems could be the usage of the gate for other aircraft. This is why the aircraft needs to be parked at another position which results in higher costs.

Depending on the network structure, the first approach is not recommendable for a schedule longer than one week if most flights are not connected to a maintenance station. In this case it does not assure a regularity for the maintenance availability. Furthermore, a minimal duration for maintenance availability should be set to avoid too short stays. Nevertheless, the number of needed constraints is smaller and as a consequent also the run time. The second approach is closer to the reality because most airlines plan regular maintenance events to avoid long downtimes of the aircraft. Contrariwise, the longer ground times during the day are unlikely for airlines because of the extra effort.

4. Conclusion and outlook

This paper presents two approaches to generate aircraft rotations including maintenance availability in a simplified manner and compares them regarding their suitability within the DAT. The first one makes sure that a minimal total maintenance availability per rotation is met while the other one limits the days between two successive nights at a maintenance station. Due to the run time, the first approach turned out to be the more suitable for the DAT despite the irregularity of the availability for maintenance. Because of the possibility to give feedback, the next tool would validate the regularity if it is not suiting the required regulations. The other approach has the advantage that such a validation would not be needed so that it can be used as an alternative in cases where the runtime is not a limiting factor.

In the further development of the DAT, the aircraft rotation model should give feedback to the fleet assignment tool if it is not possible to create rotations. This feedback could include flights which can not be flown by the same type or a list of flights which can not be integrated into rotations of one type. Even if a type is assigned to too many flights, this is given as feedback. Moreover, feedback from the next tools should be included as additional constraints. This not only includes invalid cases but also the ones to get more optimal rotations e.g. a crew which could stay at an aircraft if the rotation would be changed. This way, the overall result of the DAT could be closer to optimal than results without feedback.

In the future, the algorithms will be further improved. To describe the airlines more realistically, more functionalities can be added to the model. For example some airlines assign aircraft to a specific airport where the machines typically stay overnight. Problems arise if this airport has no maintenance facilities. This issue can be solved by regularly swapping such an aircraft with one assigned to a maintenance station. Furthermore, in the first approach the minimal length of a maintenance arc should be limited to avoid too short stays without maintenance possibility. In the second approach the ground time during the day could be limited to avoid unused aircraft. Furthermore, the run time may be reduced if constraints were added to minimize the calculation of symmetric solutions which generate the same rotations in a different order. These are no new solutions because the objective value is not different but depending on the number of rotations calculated they could have an impact on the run time.

References

Airbus, 2025. Aircraft characteristics. URL: aircraft.airbus.com. Accessed: 16 Sep 2024.

Al-Thani, N.A., Ben Ahmed, M., Haouari, M., 2016. A model and optimization-based heuristic for the operational aircraft maintenance routing problem. Transportation Research Part C: Emerging Technologies 72, 29–44.

Boeing, 2025. Airplane characteristics for airport planning. URL: www.boeing.com. Accessed: 16 Sep 2024.

Cirium, 2025. Fleets Analyzer. URL: `www.cirium.com/products/views/fleets-analyzer/`. Accessed: 21 Jul 2024.

Clarke, L., Johnson, E., Nemhauser, G., Zhu, Z., 1997. The aircraft rotation problem. Ann. Oper. Res. 69, 33–46.

Eltoukhy, A., Chan, F., Chung, S., Niu, B., Wang, X., 2017. Heuristic approaches for operational aircraft maintenance routing problem with maximum flying hours and man-power availability considerations. Industrial Management & Data Systems 117, 2142–2170.

Gurobi Optimization, LLC, 2024. Gurobi optimizer reference manual. URL: `www.gurobi.com`.

Khaled, O., Minoux, M., Mousseau, V., Michel, S., Ceugniet, X., 2018. A compact optimization model for the tail assignment problem. European Journal of Operational Research 264, 548–557.

Lagos, C., Delgado, F., Klapp, M.A., 2020. Dynamic optimization for airline maintenance operations. Transp. Sci. 54, 998–1015.

Liang, Z., Chaovalitwongse, W.A., 2013. A network-based model for the integrated weekly aircraft maintenance routing and fleet assignment problem. Transp. Sci. 47, 493–507.

Liang, Z., Chaovalitwongse, W.A., Huang, H.C., Johnson, E.L., 2011. On a new rotation tour network model for aircraft maintenance routing problem. Transp. Sci. 45, 109–120.

Mirjafari, M., Alireza, R.K., Shoja, A., 2023. Aircraft routing problem considering various maintenance operation factors: A literature review. Journal of Industrial Engineering International 19, 1–13.

Parmentier, A., Meunier, F., 2020. Aircraft routing and crew pairing: Updated algorithms at air france. Omega 93, 102073.

Ruan, J.H., Wang, Z.X., Chan, F.T.S., Patnaik, S., Tiwari, M.K., 2021. A reinforcement learning-based algorithm for the aircraft maintenance routing problem. Expert Syst. Appl. 169, 114399.

Röhrs, M., Albrecht, S., Ehlers, T., Lütjens, K., 2025. Automated repair of input data for fleet assignment in a digital airline twin. Accepted.

Sabre GLBL Inc., 2025. Sabre Market Intelligence - database. URL: www.sabre.com. Accessed: 23 Apr 2025.

Safaei, N., Jardine, A.K., 2018. Aircraft routing with generalized maintenance constraints. Omega 80, 111–122.