Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 52 (2021) 445–452

Transportation Research Procedia

www.elsevier.com/locate/procedia

23rd EURO Working Group on Transportation Meeting, EWGT 2020, 16-18 September 2020, Paphos, Cyprus

# The Tail Assignment Problem: A Case Study at Vueling Airlines

Manuel Fuentes[a], Luis Cadarso[b,*], Vikrant Vaze[c], Cynthia Barnhart[d]

[a]*Cisneria Engineering, Spain.*

[b]*Aerospace Systems and Transport Research Group, European Institute for Aviation Training and Accreditation (EIATA), Rey Juan Carlos University, Madrid, Spain.*

[c]*Thayer School of Engineering, Dartmouth College, USA.*

[d]*Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, USA.*

**Abstract**

To schedule aircraft, assignments of fleet types to flights and of aircraft to routes must be determined. The former is known as the fleet assignment problem while the latter is known as the aircraft routing problem in the literature. Aircraft routing is usually addressed as a feasibility problem whose solution is needed for constructing crew schedules. These problems are usually solved from 4 to 6 months before the day of operations. Therefore, there is limited information regarding each aircraft's operational condition. The tail assignment problem, which has received limited attention in air transportation literature, is solved when additional information regarding operational conditions is revealed aiming at determining each aircraft's route for the day of operations accounting for the originally planned aircraft routes and crew schedules. Therefore, it is a problem to be solved the day before operations. We propose a mathematical programming approach based on sequencing that captures all operational constraints and maintenance requirements while assignment costs are minimized. Computational experiments are based on realistic cases drawn from a Spanish airline, which features a network with more than 1000 flights and more than 100 aircraft.



*Keywords:* aircraft, tail, routing, airline, maintenance, air transportation.

## 1. Introduction

Tail assignment is the step in an airline planning process where specific aircraft (tails) are assigned to flights in a given flight schedule. This assignment is performed a few days, or even one day, prior to operation and is subject to multiple constraints, including maintenance and operational constraints, among others. This problem is addressed in

* Corresponding author. Tel.: +34- 91 488 8775.
*E-mail address:* luis.cadarso@urjc.es


10.1016/j.trpro.2021.01.052

an operational planning horizon for two reasons. First, weeks before the day of operations, little to no information is known regarding maintenance needs for each tail, which is the main reason for considering generic maintenance opportunities in the aircraft routing problem. Second, late changes in the flight schedule may also arise because some flights may be rescheduled or canceled, and some others may be newly introduced before the operation.

The aircraft routing problem has been researched for several decades. Given the assignment of fleet types to flights, it entails determining weeks to months ahead of operations the sequence of flights to be flown by each aircraft and ensuring each flight is flown exactly once, each aircraft visits maintenance stations at regular intervals to make generic maintenance opportunities available, and only available aircraft are utilized (Desaulniers et al., 1997; Clarke et al., 1997; Barnhart et al., 1998). Note that, it is common practice to generate sequences of flights for aircraft very early in the planning process because of the need to provide input data to the crew planning process and to plan long-term maintenance. Sequences of flights have been named in several different ways in this context; for example, 1-day routes (also known as rotations), strings, and lines-of-work (also known as big-cycles). These names usually refer to different business models (Lacasse-Guay et al., 2010).

As introduced before, the tail assignment problem, which usually differs considerably between airlines, is addressed very close to the day of operations. Here, all individual operational constraints are considered. The problem is solved for a time horizon, which usually spans several days, and provides fully operational assignments of aircraft to sequences of flights. In practice, the input sequences of flights, which are determined in the aircraft routing problem, are usually not suitable for satisfying operational constraints on the day of operations and they must be updated (Liang et al., 2015). Some variants of this problem have been developed in response to different airline business practices and involving planning horizons from months to days (Maher et al., 2018). Grönkvist (2005) and Gabteni and Grönkvist (2009) have analyzed the problem and proposed an approach to aircraft assignment that captures many operational constraints and finds initial solutions quickly. However, maintenance is not comprehensively modeled, and their computational results drawn from real-world instances of medium-scale tail assignment problems show that near-optimal solutions are only obtained when longer running times are allowed. Khaled et al. (2018) investigated a compact mathematical programming formulation for the problem to minimize the cost of operating flights and maintenance while complying with several operational constraints. They showed computational experiments on instances featuring up to 40 aircraft and 1,500 flights that connect 21 airports. Maher et al. (2018) focused on the analysis of the problem by employing the 1-day routes approach and developed a tail assignment model. To model maintenance for the following days, they introduced look-ahead constraints. But they stated maintenance misalignments in the solutions.

This research presents a mathematical model and a solution approach for the tail assignment problem to address large-scale real-world instances of the problem. The mathematical formulation captures all the operational requirements at the airline under consideration, and it goes one step further by including apron-related requirements. Existing approaches in the literature do not successfully capture such a level of detail. To solve large-scale instances a rolling horizon method is employed in which the model is solved in several smaller sub-models. This method is useful for obtaining feasible solutions to the problem within short computational times. The obtained solution is then improved by means of a heuristic-based approach, which turns out to be able to obtain the optimal solution. Overall, the proposed methodology identifies optimal solutions to large-scale real-world problem instances within short computational times. Finally, the results of realistic computational experiments using data from Vueling Airlines, one of the main airlines in Spain, are presented.

## 2. Problem description

### 2.1. Rotations & Lines-of-work

A rotation is a sequence of flights to be performed by the same tail during a single day. A sequence of rotations to be performed by the same tail during a certain period comprising several days is referred to as a Line-of-work (*LoW*). Note that rotations and *LoW*s are outputs of the aircraft routing problem, which are obtained after the information regarding flight schedules and fleet assignment is known. The aircraft routing problem in the planning phase assumes that all rotations each day start and end at the same airport, assigning a base to each single tail. In tail assignment, rotations and the fleet assigned to them are the main inputs, and they should be maintained as much as possible when allocating tails to flights because crew schedules are based on them. However, the tail assignment process may not stick to planned rotations in case they become infeasible. When allocating tails to rotations, the maintenance needs of tails must be taken into consideration, because otherwise assignments may be infeasible. For example, it is frequently the case where assignments involve rotations ending in a base and tails requiring maintenance in another base. To deal with this situation, swaps are introduced. A swap is the recombination of two rotations with different bases that allows the tails allocated to them to change their bases. Consequently, a tail with a certain maintenance need can finish the day at the base where tasks must be performed. Figure 1 shows a base swap. One tail is allocated to a rotation that is based in Madrid. If this tail requires a maintenance task that must be performed in Barcelona, it must be swapped. Swaps must comply with some rules, such as those requiring space–time compatibility. In addition, their impact on crews must be limited and therefore, only a restricted set of them, which is identified when the aircraft routing is solved, is permitted.

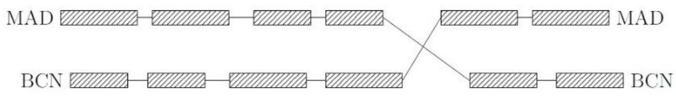


Fig. 1. Example of Madrid and Barcelona bases swap.

### 2.2. Maintenance activities

Various types of maintenance activities must be performed on an aircraft, which may or may not be scheduled ahead of time. Scheduled tasks are known in advance, as they must be performed periodically, whereas some others result from operations, usually after failures in certain equipment. Based on their periodicity, they can be categorized as follows:

- Flight line checks, which are minor daily checks.
- Overnight checks, which are minor checks performed every 2 days during nights.
- A checks, defined as light checks performed every few hundred flying hours;
- B checks, which are light checks performed every few months.
- C checks, defined as heavy checks performed approximately every 2 years.
- And D checks, which are heavy checks performed every 6–10 years.

The objective of these checks is to conduct both routine and nonroutine maintenance of the aircraft. Maintenance includes scheduling the repair of known problems; replacing items after a certain number of flight hours, cycles, or calendar time; repairing previously discovered defects (e.g., reports logged by pilot or crew, line inspection, items deferred from previous maintenance), and performing scheduled repairs. When performing tail assignment, some maintenance checks (hereafter referred to as maintenance tasks, maintenance activities, or maintenance duties) are more flexible than others. As an example, an A check can be scheduled for a day but can be delayed by a few days if the tail has remaining flying hours.

### 2.3. Allocation rules

The allocation of tails to flights is subject to several rules. Some of these rules are operational, and they can be considered as either hard or soft restrictions. Some others are related to business considerations, such as the preference for certain tails due to their capacity or efficiency. For ease of exposition, all the allocation rules are categorized into two groups: hard restrictions and soft restrictions. We consider the hard restrictions to be all the allocation rules that must be respected under any condition; they can never be violated. Soft restrictions refer to all the allocation rules that can be violated but have a certain impact on the solution quality. The number of existing rules is huge, and they are complex, which makes this problem challenging for planners.

Examples of hard restrictions are as follows:

- An aircraft type is not allowed to fly from/to an airport.
- A certain tail cannot operate at a certain airport during certain time periods due to noise limitations.
- An aircraft type cannot fly from/to an airport without maintenance resources. Hiring local and/or external sources is considered dangerous. A minor failure could lead to a tail being declared grounded.

Examples of soft restrictions are as follows:

- An aircraft type has limited performance at a certain airport due to maximum takeoff weight. If the tail is not at its maximum weight, it can fly from that airport.
- Certain tails can fly from/to some airports only if the airline pays a penalty (e.g., noise pollution).
- A rotation is to be assigned to any tail belonging to the fleet as decided in the fleet assignment problem. However, tails belonging to different fleets could also be assigned.
- Some specific tails within a fleet are more efficient (e.g., lower fuel consumption because of newer engines) than others when flying some flights.
- A tail should not fly to an airport without maintenance capabilities because it could lead to a grounded aircraft. This requisite resembles the last example in the list of hard restrictions examples but featuring lower priority.

### *2.4. Objectives*

There are several objectives in the tail assignment problem. In fact, they might vary as the revealed information evolves as the day of operations gets closer. In the early stages, the main objective is feasibility, whereas during the later stages, meeting optimization criteria acquires importance. The following factors are among the most crucial ones:

- Violations of soft restrictions: minimizing these violations is desirable not only because of the economic benefit (e.g., avoiding penalty fees) but also because of aircraft availability (e.g., flying certain tails to an airport without its own maintenance facilities increases the probability of having a grounded aircraft).
- Fleet changes: their minimization means minimizing load factors changes, which may impact profitability of flights and airline image because of denied boardings.
- Use of swaps: minimizing their use increases operations predictability because swaps may propagate delays due to crew schedules, which cannot be always swapped the same way as tails can be.
- Buffer times: maximizing the time between rotations and modifying scheduled rotations to add buffer times after flights with expected delays may increase schedule punctuality.
- Fuel cost: although this cost is mainly fixed after solving the fleet assignment problem, it can be further minimized because depending on engines, some tails in a fleet consume more fuel when flying short flights, whereas some others do so on long flights.
- Apron optimization: minimizing conflicts while maneuvering during departures from the apron reduces delays increasing schedule punctuality. Note that a conflict takes place when two aircraft depart from parking spots that are close to each other in the apron during a predefined time period.

## 3. Mathematical model

The model we propose is an Integer Linear Programming (ILP) model. It aims at determining rotations and *LoWs* for a set of tails flying a given flight and maintenance schedule. Its mathematical formulation is based on a structure in which the tasks to be assigned are considered nodes and the connections between them arcs (edges). The employed sets, parameters, and variables in the model are defined first. Then, the mathematical formulation is described.

### *3.1. Sets*

- T is the set of tasks indexed by $i$, $j$ and $k$. Each task is defined by its origin and destination airports and its start and end times. Tasks can be either flights or maintenance activities. Note that maintenance activities have the same origin and destination airports.
- A is the set of tails indexed by $a$.
- B is the set of bases (airports) indexed by $b$.
- C is the set of all potential conflicts in apron indexed by $c$.
- D is the set of days indexed by $d$.
- F is the set of fleet types indexed by $f$.
- $T_a \subseteq T$ is the set of tasks that are compatible with tail $a$.

- $T_a^s \subseteq T$ is the set of tasks that can be the first task assigned to tail $a$ in the period under study.
- $T_i^+ \subseteq T$ is the set of tasks that can be assigned right after task $i$ in an $LoW$.
- $T_i^- \subseteq T$ is the set of tasks that can be assigned right before task $i$ in an $LoW$.
- $T^o \subseteq T$ is the set of tasks that are maintenance tasks of type "overnight check".
- $T^n \subseteq T$ is the set of tasks that are nocturnal flights, which are defined as flights flying during the night.
- $T_{i+}^n \subseteq T^n$ is the set of tasks that are nocturnal flights and belong to the next night of nocturnal flight $i$.
- $T_d \subseteq T$ is the set of tasks that are flights and belong to day d.
- $A_i \subseteq A$ is the set of tails that are compatible with task $i$.
- $A_f \subseteq A$ is the set of tails that belong to fleet type $f$.

### 3.2. Parameters

- $w_a^i \in \mathcal{R}^+$, defined for $i \in T$, $a \in A_i$ to indicate the cost of assigning task $i$ to tail $a$.
- $g_a^{i,j} \in \mathcal{R}^+$, defined for $i \in T, j \in T_i^+$, $a \in A_i \cap A_j$ to indicate the cost of assigning consecutively tasks $i$ and $j$ to tail $a$.
- $u^i \in \mathcal{R}^+$, defined for $i \in T$ to indicate the cost of task $i$ not being assigned to any tail.
- $h^c \in \mathcal{R}^+$, defined to indicate the cost of having conflict $c$.
- $p_a^b \in \{0,1\}$, defined for $a \in A$ and $b \in B$ to indicate whether tail $a$ is originally positioned in base $b$.
- $q_{i-}^b \in \{0,1\}$, defined for $i \in T$ and $b \in B$ to indicate whether task $i$ ends at base $b$.
- $q_{i+}^b \in \{0,1\}$, defined for $i \in T$ and $b \in B$ to indicate whether task $i$ starts from base $b$.
- $r_f^d \in \mathcal{R}^+$, defined for $f \in F$ and $d \in D$ to indicate the maximum number of tails of fleet type $f$ that can actually fly on day $d$.
- $s_c^{i,a} \in \{0,1\}$, defined to indicate whether the assignment of task $i$ to tail $a$ is involved in apron-related conflict $c$.
- $M$ is a big enough number to be an upper bound to the number of assigned tasks to any tail on any day.

### 3.3. Variables

- $\alpha_i^a \in \{0,1\}$, defined for $i \in T$, $a \in A_i$, is 1 if task $i$ is assigned to tail $a$ and 0 otherwise.
- $\iota_i^a \in \{0,1\}$, defined for $i \in T_a^s$, $a \in A$, is 1 if task $i$ is the first task assigned to tail $a$ and 0 otherwise.
- $\gamma_{i,j}^a \in \{0,1\}$, defined for $i \in T, j \in T_i^+$ and $a \in A_i \cap A_j$, is 1 if tasks $i, j$ are both assigned to tail $a$ and they are performed consecutively and 0 otherwise.
- $\upsilon_i \in \{0,1\}$, defined for $i \in T$, is 1 if task $i$ remains unassigned and 0 otherwise.
- $\nu_d^a \in \{0,1\}$, defined for $d \in D$, $a \in A$, is 1 if tail $a$ has been assigned any flights on day $d$ and 0 otherwise. Note that maintenance duties are not considered here.
- $\sigma_c \in \{0,1\}$, defined for $c \in C$, is 1 if conflict $c$ in apron is present in the solution and 0 otherwise.

### 3.4. Objective function

$$z = \sum_{i \in T} \sum_{a \in A_i} w_a^i \alpha_i^a + \sum_{i \in T} \sum_{j \in T_i^+} \sum_{a \in A_i \cap A_j} g_a^{i,j} \gamma_{i,j}^a + \sum_{i \in T} u^i \upsilon_i + \sum_{c \in C} h^c \sigma_c \qquad (1)$$

The objective function in (1) is the sum of four terms, which are as follows and in the following order: cost for task–tail combinations, cost for task connections, penalties for not covering tasks, and penalties for all apron-related conflicts. Next, the sets of constraints in the model are presented. We categorize the constraints into five main types.

### 3.5. Task covering and sequencing constraints

$$\sum_{a \in A_i} \alpha_i^a + \upsilon_i = 1 \qquad \forall i \in T \qquad (2)$$

$$\sum_{i\in T_j^-\cap T_a} \gamma_{i,j}^a + \iota_i^a = \alpha_i^a \qquad \forall j \in T, a \in A_j \tag{3}$$

$$\sum_{k\in T_j^+\cap T_a} \gamma_{i,j}^a \leq \alpha_i^a \qquad \forall j \in T, a \in A_j \tag{4}$$

Constraints (2) state that each task is assigned to one tail or it remains unassigned. Constraints (3) and (4) are sequencing constraints. The former specify that a specific task assigned to a tail must either be preceded by just one other task or be the first one in that *LoW*. The latter constraints state that a task assigned to a tail can be succeeded by at most one other task.

### 3.6. Line-of-Work constraints (LoW)

$$\sum_{i\in T_a^s} \iota_i^a \leq 1 \qquad \forall a \in A \tag{5}$$

Constraints (5) are the *LoW* initialization constraints; each tail has at most one first task. Note that a tail may not be used at all. There will not be any *LoW* initialization for it in such a case.

### 3.7. Overnight checks constraints

$$\alpha_i^a + \alpha_j^a \leq 1 \qquad \forall i \in T^n, \forall j \in T_{i+}^n, \forall a \in A_i \cap A_j \tag{6}$$

$$p_a^b \iota_j^a + \sum_{i\in T_j^-\cap T_a} q_{i-}^b \gamma_{i,j}^a \geq \sum_{k\in T_j^+\cap T_a} q_{k+}^b \gamma_{j,k}^a \qquad \forall j \in T^o, \forall a \in A_j, \forall b \in B \tag{7}$$

Some of the overnight checks (a type of maintenance task described in Section 2.2, also referred as "daily checks") can be missing from the set of tasks to be performed. This means that they are not always explicitly stated as maintenance tasks by the airline, but they still must be done. This is modeled by constraints (6), which requires that any tail should be prevented from flying consecutive nights to ensure that every tail is resting overnight every 2 days. Also, daily checks are special in that they have no base associated with them, as it is possible to perform such checks at any airport. A way of modeling this particularity (for those which are explicitly in the set of maintenance tasks) is by including in the model as many copies of each daily check task as airports in the network and limiting the assignment to one of them. However, we are modeling this by adding the constraints (7), which state that either the daily check task is the first one in the *LoW* or it is preceded and followed by tasks ending and beginning from the same airport.

### 3.8. Backups constraints

$$\sum_{i\in T_a\cap T_d} \alpha_i^a = M\nu_d^a \qquad \forall a \in A, \forall d \in D \tag{8}$$

$$\sum_{a\in A_f} \nu_d^a \leq r_f^d \qquad \forall f \in F, \forall d \in D \tag{9}$$

Constraints (8) and (9) model the backup tails. The former constraints hold that a tail is ordinary (that is, not backup) each day if it has flights assigned that day. The latter limit the number of ordinary tails per day.

### 3.9. Apron constraints

$$\sum_{i\in T}\sum_{a\in A} s_c^{i,a} \iota_i^a \leq 1 + \sigma_c \qquad \forall c \in C \tag{10}$$

Conflicts in stands are modeled by constraints (10). Note that the number of conflicts is not limited but each conflict is penalized in the objective function.

## 4. Solution approach and computational experiments

Real-world instances of the tail assignment problem usually involve millions of variables, which results in a time-consuming process of solving them. Although this problem is a planning problem and the solution time is not as critical as in operations-related problems, reducing the solution time increases planners' flexibility. We found that commercial branch-and-cut and heuristics approaches failed to provide optimal solutions to large-scale problems within affordable computational times. To efficiently solve the problem, we have developed and applied an algorithm based on rolling horizon methods (Sethi and Sorger, 1991) to obtain solutions quickly. The Rolling Horizon Algorithm (RHA) is based on rolling horizon methods for solving mixed 0-1 deterministic optimization problems. It entails solving a sequential series of integer programming subproblems such that the variables in each one are partitioned into three subsets. The values of the variables in the first subset are fixed to values obtained in previous solutions, the 0-1 variables in the second subset are kept free, and the values of the variables in the third subset are fixed to value 0. RHA partitioning is based on days. But, the RHA by itself cannot prove optimality. To prove it, a different approach should be used. For that purpose, the solution provided by that algorithm can be used as an initial solution and the entire problem solved. Although exact methods should be used to guarantee optimality, we have empirically discovered that feeding the "solution polishing" heuristic provided by CPLEX with the initial solution obtained by the RHA provides the optimal solution.

We evaluated the model's performance with case studies focusing on realistic instances drawn from Vueling Airlines. The data were provided by the airline and represent its operations in Europe for the year 2019. The data set consists of operating schedule information, operating expenses, passenger demand values, BCN airport apron layout, maintenance capacities, and the available fleet for the time period from October 6 to October 10 2019. The air network is a mixed hub-and-spoke and point-to-point network with 173 airports spread throughout Europe and with some in Asia and Africa. Approximately 700 flights operate within the network on a representative day, and three different fleet types were available in this case study. To determine the behavior of the model and solution approaches for real-world instances, a set of five case studies were proposed. All of them were based on the same period—October 2019—but featured different planning horizons, ranging from 1 day to 5 days. Number of flights and maintenance activities ranged from 682 to 3,279 and from 59 to 100, respectively, all of them operated with a set of 127 tails. Mathematical model sizes ranged from 117,217 to 2,731,427 binary variables and from 120,34 to 581,291 constraints.

We first solved all the case studies using the branch-and-cut and heuristic approach available in the commercial solver IBM ILOG CPLEX 12.9.0. These results are shown in Table 1. Each row in Table 1 describes a case study, and each is identified by the number of days in the planning horizon, the number of flights, the number of maintenance activities, and the number of tails. Additionally, for each case study, Table 1 shows the lower bound (L.B.), the incumbent solution (I.S.), the optimality gap (O.G.), and the computational time in seconds (T.). The lower bound is the highest bound provided by the solver. The incumbent solution is the best solution found. The optimality gap is the relative gap of the incumbent solution with reference to the lower bound. The computational time is the time the solver ran. As the problem size increased, computational time exponentially increased, meaning that this solution approach was not able to yield solutions within a reasonable time if the period to be solved for was longer than a few days. Recall that this problem must be solved every day and that the solution must be implemented by the next day.

Table 1. Solutions of all the case studies using the branch-and-cut and heuristics approach in the commercial solver IBM ILOG CPLEX 12.9.0.

| No. of days | Flights | Activities | Tails | L.B: | I.S. | O.G. (%) | T. (s) |
|---|---|---|---|---|---|---|---|
| 1 | 682 | 59 | 127 | 193.05 | 193.05 | 0.00 | 1.05 |
| 2 | 1,376 | 77 | 127 | 333.02 | 333.02 | 0.00 | 7.88 |
| 3 | 1,999 | 86 | 127 | 451.52 | 451.52 | 0.00 | 1,200.27 |
| 4 | 2,626 | 95 | 127 | 599.35 | 599.35 | 0.00 | 47,312.28 |
| 5 | 3,279 | 100 | 127 | 728.10 | 18,690.10 | 96.10 | 86,403.72 |

Because the previous approach failed to provide high-quality solutions within reasonable computational times when the problem size increased, the solution approach previously described is tested, which consists in feeding the "solution polishing" heuristic provided by CPLEX with the initial solution obtained by the RHA. Table 2 presents the following information for the case studies featuring planning horizons of 4 and 5 days. Different rows correspond to different combinations of solution approach (in the first column) and number of days in the planning horizon (in the second column). The third column shows the number of unassigned tasks (U.), which are either flights or maintenance activities, the fourth column shows the incumbent solution (I.S.), the fifth shows the optimality gap (O.G.), and the last column shows the computational time in seconds (T.). The number of unassigned tasks is regarded as the key performance indicator, and if it is not zero, the schedule must be changed, which is time consuming and risky due to the nature of the problem being solved. The main conclusion to draw from Tables 1 and 2 is that the RHA+SP approach is superior; it can provide the optimal solution in a significantly shorter computational time compared to the branch-and-cut and heuristics approaches used by commercial solvers.

Table 2. Comparison of the performance of the presented solution approaches for the case studies featuring 4 and 5 days in the planning horizon.

| Approach | No. of days | U. | I.S. | O.G. (%) | T. (s) |
|---|---|---|---|---|---|
| RHA | 4 | 8 | 1,407.02 | 57.40 | 78.55 |
| RHA+SP | 4 | 0 | 599.35 | 0.00 | 279.50 |
| RHA | 5 | 6 | 1,376.52 | 47.11 | 167.90 |
| RHA+SP | 5 | 0 | 728.10 | 0.00 | 407.42 |

## 5. Conclusions

We have approached the tail assignment problem capturing a wide variety of details and providing a framework for obtaining optimal plans instead of just producing feasible solutions. Among the details considered, all the relevant aircraft maintenance-related constraints and flight operation requirements are included. In addition, potential conflicts during aircraft taxi operations in aprons are also considered, such that departures are optimally decided to avoid multiple aircraft departing at the same time from the same location. Note that existing approaches in the literature do not successfully capture such a level of detail. And attempts to reach this level, while trying to also provide exact solutions, have failed. With the methodology we have developed, we were able to solve real-world instances in short computational times while proving the optimality. The algorithms we have developed for solving the problem are based on two main phases. First, a feasible solution is obtained by employing a rolling horizon method. Second, a heuristic-based approach improves the obtained feasible solution. We presented results of computational experiments using data from Vueling Airlines, one of the major Spanish airlines.

## Acknowledgements

This research was supported by Project Grants TRA2016-76914-C3-3-P by the *Ministerio de Economía y Competitividad*, Spain, and CAS19/00036 by the *Ministerio de Ciencia, Innovación y Universidades*, Spain.

## References

Barnhart, C., Boland, N. L., Clarke, L. W., Johnson, E. L., Nemhauser, G. L., & Shenoi, R. G. (1998). Flight string models for aircraft fleeting and routing. Transportation science, 32(3), 208-220.

Clarke, L., Johnson, E., Nemhauser, G., and Zhu, Z. The aircraft rotation problem. Annals of Operations Research, 69, 33-46, 1997.

Desaulniers, G., Desrosiers, J., Dumas, Y., Solomon, M., and Soumis, F. Daily aircraft routing and scheduling. Management Science, 43(6):841-855, 1997.

Gabteni, S. and Grönkvist, M. Combining column generation and constraint programming to solve the tail assignment problem. Annals of Operations Research, 171(1):61, 2009.

Grönkvist, M. (2005). The tail assignment problem (pp. 0328-0328). Chalmers tekniska högskola.

Khaled, O., Minoux, M., Mousseau, V., Michel, S., and Ceugniet, X. A compact optimization model for the tail assignment problem. European Journal of Operational Research, 264(2):548–557, 2018.

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

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

Maher, S., Desaulniers, G., and Soumis, F. The daily tail assignment problem under operational uncertainty using look-ahead maintenance constraints. European Journal of Operational Research, 264(2):534–547, 2018.

Sethi, S. and Sorger, G. A theory of rolling horizon decision making. Annals of Operations Research, 29(1):387–415, 1991