A Consecutive Flight Leg Model for the Aircraft Maintenance Routing Problem

Ida Gjergji[1], **Lucas Kletzander**[2], **Hendrik Bierlee**[3, 4, 5], **Nysret Musliu**[2], **Peter J. Stuckey**[3, 4]

[1]DBAI, TU Wien, Vienna, Austria

[2]Christian Doppler Laboratory for Artificial Intelligence and Optimization for Planning and Scheduling, DBAI, TU Wien, Vienna, Austria

[3]Monash University, Melbourne, Australia

[4]OPTIMA ARC Industry Training and Transformation Centre, Melbourne, Australia

[5]KU Leuven, Leuven, Belgium

{ida.gjergji,lucas.kletzander,nysret.musliu}@tuwien.ac.at, henk.bierlee@kuleuven.be, peter.stuckey@monash.edu

Abstract

The aircraft maintenance routing problem with a maintenance distribution objective (AMRP-D) considers a set of flight legs that have to be assigned to a set of available aircraft. While the start and end time of the flight legs are specified in advance, the aircraft are required to undergo various maintenance tasks in the planning horizon. Scheduling these maintenance tasks as evenly as possible is the objective of the AMRP-D, however, for most instances it is even difficult to find a feasible solution covering all maintenance requirements. This paper presents a new constraint programming model for the stated problem in order to provide feasible solutions for small instances. The proposed model is tested on the available benchmark instances, resulting in more feasible solutions.

Introduction

The aircraft maintenance routing problem (AMRP) is one of the main optimization problems emerging in the airline industry. The AMRP deals with a set of flight legs and a set of aircraft available to operate these flights. In this case, the start and the end times of the flight legs are already known. The aircraft are required to carry out these flight legs together with certain maintenance tasks. The scheduling of the maintenance tasks has to be done in accordance with regulations imposed on the aircraft. A comprehensive overview of work in this area can be found in Eltoukhy, Chan, and Chung (2017), Temucin, Tuzkaya, and Vayvay (2021), and Xu, Wandelt, and Sun (2023).

A new version of this problem, AMRP-D (AMRP with a maintenance distribution objective), was introduced by Kletzander, Gjergji, and Musliu (2024). The AMRP-D stems from a real-world application, and it focuses on the even distribution of the maintenance tasks. This objective enables the possibility of aircraft schedules with better utilization of maintenance resources, avoiding both over- and under-utilization, which in the traditional AMRP are disregarded.

The first solution methods for the AMRP-D were delivered by Gjergji et al. (2025): a constraint programming (CP) model and a decomposition approach. The decomposition approach considers slices of the scheduling period by allocating each flight leg based on its time of occurrence to a single slice. Then the CP model is used in every slice for the assignment of the flight legs to the aircraft. In this manner, the decomposition approach is able to solve more instances in comparison to the standalone CP model. Gjergji et al. (2025) also present a Simulated Annealing (SA) approach and a Large Neighborhood Search (LNS) to optimize the objective function of the AMRP-D, where SA provides the best results. Since obtaining a feasible solution is hard for this problem, and there are still instances where no feasible solution has been found, in this paper, we present an alternative CP model for the provision of feasible solutions for the AMRP-D. The goal is to investigate if the performance of the model can be enhanced when consecutive flight legs are explicitly modeled, replacing optional variables with regular variables in the process. The model is tested on the available benchmark instances, where the number of feasible solutions is increased compared to the previous model.

Problem Description

The Aircraft Maintenance Routing Problem (AMRP) is specified by a set of flight legs, regulations for maintenance, and a set of aircraft including their recent flight and maintenance history. All times are given as integers.

Flight Legs

A set of n flight legs $T = \{t_1, \ldots, t_n\}$ is given, each leg t_i is associated with a start time s_i, an end time e_i, and a flight time f_i (with $f_i < e_i - s_i$, as the flight leg contains both the flight time and the preparation/turnaround times). As such, flight legs are not allowed to overlap for the same aircraft, but there is no minimum distance between legs on the same aircraft. The first m legs in T denote the last leg of each of the m aircraft from the previous scheduling period which are fixed to the particular aircraft. Note that in general, each flight leg is also associated with a start location and an end location, defining the routing aspect of the problem. However, in this paper we deal with a version using a single home base where outbound and following inbound flights are fused to one flight leg (with no maintenance at the destination). Therefore, all locations are equal and omitted.

Maintenance

Aircraft require different types of maintenance. Each maintenance type takes a different amount of time to complete.

A maintenance on an aircraft cannot overlap with its other maintenance or flight legs. Some maintenance types compete for a single hangar. Furthermore, some types are due periodically (regardless of flight time), while others are due based on the cumulative flight time. The following types of maintenance are used in the AMRP-D:

- Regular: An aircraft can fly for at most $regl = 47$ hours after the end of the previous regular maintenance, then it needs regular maintenance taking $regd = 2.5$ hours before taking off again.
- Weekly: An aircraft can fly for at most $weekl = 156$ hours (6.5 days) after the end of the previous weekly maintenance, then it needs weekly maintenance taking $weekd = 7$ hours before taking off again. Weekly maintenance includes regular maintenance.
- Major: There are four different types of major maintenance. Each of them is independent from the others. Each follows the same rule regarding time: After at most $majl$ = 950 hours of cumulative flight time since the last maintenance of the same type, the aircraft needs major maintenance taking $majd = 14$ hours before taking off again. Each type of major maintenance includes regular and weekly maintenance. There are further differences regarding subtypes:
  - `MH1` and `MH2`: These require a single hangar, meaning that only one aircraft can perform any of these two types of maintenance at once.
  - `MR1` and `MR2`: These two types do not require the hangar.

Aircraft

A set of m aircraft $A = \{a_1, \ldots, a_m\}$ is given, each aircraft a_j is preassigned to the history flight leg t_j, and is associated with the time before the start of the planning horizon that the last regular maintenance ended r_j, the time before the start of the planning horizon that the last weekly maintenance ended w_j, and the cumulative flight time since each of the major maintenance types $majp_{kj}$ with $k \in K$ and $K = \{1, 2, 3, 4\}$, where 1 and 2 correspond to `MH1` and `MH2`, while 3 and 4 correspond to `MR1` and `MR2`.

Solution

The legs for $j \in \{1, \ldots, m\}$ are fixed to the corresponding aircraft. A feasible solution assigns all remaining flight legs to aircraft, and the required types of maintenance to specific aircraft and time intervals, such that:

- No overlapping flight legs are assigned to any aircraft.
- No maintenance intervals are violated.
- At most one aircraft is assigned `MH1` or `MH2` at any point in time.

The objective of the AMRP-D is to minimize $\sum_s m_s^2$, where m_s shows the total number of aircraft in any type of maintenance for each minute s in the planning period. A second requirement is to schedule the major maintenance tasks as late as possible, without violating feasibility. However, these objectives are not part of the stage of the problem we deal with in this paper.

MiniZinc Model

The decision variables used in the proposed model are as follows:

- T': the set of flight legs excluding the previously assigned legs $T' = T \setminus \{t_j \mid j \in A\}$.
- $tail_i$: the aircraft leg i is assigned to.
- $alloc_{ji}$: binary variable taking the value 1 if leg i is assigned to aircraft j, and 0 otherwise.
- $prev_i$: the precedent of leg i on the same aircraft.
- ft_i: total flight time flown by the aircraft up to leg i.
- mm: maximum number of maintenance tasks that can be scheduled for any aircraft j. Equation (1) defines the number mm as:

$$mm = \frac{\max_i(e_i) - start_h}{regl} + 5 \tag{1}$$

- ms_{jl}: the start time of the l^{th} maintenance for aircraft j.
- mt_{jl}: the type of the l^{th} maintenance for aircraft j, where 0 is a dummy maintenance, 1 is a regular maintenance, 2 is a weekly maintenance, and $\{3, 4, 5, 6\}$ are major maintenance tasks in the order `MH1`, `MH2`, `MR1`, and `MR2`.
- md_{jl}: the duration of the l^{th} maintenance for aircraft j.
- $majs_{jk}$: the start of major maintenance k for aircraft j.
- $majh_{jk}$: binary variable taking the value 1 if major maintenance k happens for aircraft j, and 0 otherwise.
- $majc_{jk}$: the index of the maintenance for aircraft j where major maintenance k is scheduled.

One of the main differences with the previous model by Gjergji et al. (2025) is the exclusion of optional variables, which in the earlier model covered both the selection of flight legs for aircraft and the start of maintenance tasks for aircraft. In the new model, the decision variables for the assignment of flight legs to the aircraft are expressed with the regular variable $tail$, which represents the aircraft operating each flight leg. For the start of maintenance tasks ms we use the same maximum number mm. However, we use dummy values for maintenance tasks that are not necessary in the scheduling period.

The mathematical model for the AMRP-D is displayed in Figure 1. Equation (2) assigns the previous history legs to their respective aircraft. Equation (3) denotes the first maintenance task to be the weekly history maintenance that it is completed at w_j. Equation (4) sets as a second maintenance task the regular maintenance from the history if it does not end at the same time as the weekly maintenance. Equation (5) calculates the total flight time for the legs, where the flight legs from the history account only for their flight time while the other flight legs also need the total flight of their precedent ($prev$) flight leg on the same aircraft. Equation (6) enforces that non-history flight legs should have a different $prev$ leg. Using the $prev$ variable is helpful for keeping track of the total flight time for each aircraft in order to schedule major maintenance tasks that are requested based on the accumulated flight time. Equation (7) guarantees that no flight legs are assigned to aircraft without undergoing major maintenance if their total flight time together with the cumulative

$$tail_{t_j} = j, \quad j \in A, \tag{2}$$
$$ms_{j1} = w_j - weekd \wedge mt_{j1} = 2, \quad j \in A \tag{3}$$
$$ms_{j2} = r_j - regd \wedge mt_{j2} = 1, \quad j \in A, r_j \neq w_j \tag{4}$$
$$\begin{cases} tail_i = tail_{prev_i} \wedge ft_i = ft_{prev_i} + f_i & \text{if } i \in T' \\ prev_i = i \wedge ft_i = f_i & \text{otherwise} \end{cases}, \quad i \in T \tag{5}$$
$$alldifferent([prev_i | i \in T']) \tag{6}$$
$$ft_i + majp_{k\ tail_i} > majl \rightarrow s_i \geq majs_{tail_i\ k} + majd, \quad i \in T, k \in K \tag{7}$$
$$alloc_{ji} \leftrightarrow tail_i = j, \quad j \in A, i \in T \tag{8}$$
$$disj([s_i \mid i \in T] + + \overline{ms}_j, [(e_i - s_i) \cdot alloc_{ji} \mid i \in T] + + \overline{md}_j), \quad j \in A \tag{9}$$
$$tail_i = j \rightarrow \exists_{l \in 1..mm} ms_{jl} < s_i \wedge mt_{jl} \neq 0 \wedge ms_{jl} + md_{jl} + regl \geq e_i, \quad j \in A, i \in T' \tag{10}$$
$$tail_i = j \rightarrow \exists_{l \in 1..mm} ms_{jl} < s_i \wedge mt_{jl} \neq 0 \wedge mt_{jl} \neq 1 \wedge ms_{jl} + md_{jl} + weekl \geq e_i, \quad j \in A, i \in T' \tag{11}$$
$$majh_{jk} = 0 \leftrightarrow majs_{jk} = max(e_i), \quad j \in A, k \in K \tag{12}$$
$$md = [mdx_{mt_{jl}} \mid j \in A, l \in 1..mm], \tag{13}$$
$$disj(\overline{majs}_j, [majd \cdot majh_{jk} \mid k \in K]), \quad j \in A \tag{14}$$
$$disj(\overline{majs}_1 + + \overline{majs}_2, [majd \cdot majh_{j1} \mid j \in A] + + [majd \cdot majh_{j2} \mid j \in A]), \tag{15}$$
$$mt_{jl} = 0 \rightarrow mt_{j,l+1} = 0, \quad j \in A, l \in 1..mm - 1 \tag{16}$$
$$mt_{jl} = 0 \rightarrow ms_{jl} = max(e_i), \quad j \in A, l \in 1..mm \tag{17}$$
$$\begin{cases} mt_{j\ majc_{jk}} = k + 2 \wedge ms_{j\ majc_{jk}} = majs_{jk} & \text{if } majh_{jk} = 1 \\ majc_{jk} = mm + 1 & \text{otherwise} \end{cases}, \quad j \in A, k \in K \tag{18}$$
$$mt_{jl} - 2\ in\ K \rightarrow majc_{j\ mt_{jl}-2} = l, \quad j \in A, l \in 1..mm \tag{19}$$
$$increasing(\overline{ms}_j), \quad j \in A \tag{20}$$

Figure 1: Model of AMRP-D. Concatenation of sequences is shown with $++$, disjunctive constraints are indicated with disj, rows and columns in matrices are illustrated as $\overline{ms}_j$.

flight time from the history exceeds the major limit. This is another distinction from the previous model (Gjergji et al. 2025). Equation (8) makes sure that $alloc_{ji}$ holds for each leg and its assigned aircraft. Then $alloc$ is used in Equation (9) for the disjunctive constraint to prohibit overlapping of flight legs and maintenance tasks for all aircraft. In the earlier approach (Gjergji et al. 2025), only the assigned optional variables were used for the disjunctive constraint as the constraints would hold for non-occurring assignments. Equation (10) enforces that for each aircraft maintenance tasks are scheduled in line with the regular maintenance limit (excluding dummy maintenance). Equation (11) follows the same rule for weekly maintenance limit, however regular maintenance tasks are not taken into account. Equation (12) places the major maintenance tasks that do not happen at the end of the scheduling period. Equation (13) specifies the maintenance duration depending the maintenance type mt using mdx that is set to 0 for dummy, $regd$ for regular, $weekd$ for weekly and $majd$ for major maintenance tasks. Equation (14) forces all major maintenance tasks to be different for every aircraft. Equation (15) ensures that the hangar constraint is not violated by using the disjunctive constraint for the `MH1` and `MH2` major maintenance tasks scheduled for all aircraft. Equation (16) lists the dummy maintenance tasks at the end and Equation (17) assigns the end of the scheduling period as start time for the dummy tasks. Equation (18) ensures that happening major maintenance are included in the start maintenance tasks variable ms start. For simplicity, non-occurring maintenance tasks are allocated at the $mm+1$ maintenance slot. In the model by Gjergji et al. (2025), the major maintenance tasks $majs$ were mapped into the maintenance start ms utilizing the $exists$ quantifier to ensure the maintenance slot. In our approach, we use $majc$ to allocate a maintenance slot to major maintenance that are scheduled. Equation (19) guarantees that the type of the major maintenance task is in accordance with its allocated maintenance slot. Lastly, Equation (20) enforces an increasing order for the start times of the maintenance tasks for each aircraft (duplicates are allowed due to the dummy maintenance tasks).

Experimental Evaluation

The experiments for both models were done on a computing cluster, equipped with two Intel Xeon E5-2650v4 @ 2.20 CPUs with 12 cores. CP-SAT 9.14 is used for solving the CP models as satisfaction problems (no objective function) which were modeled in MiniZinc (Nethercote et al. 2007). The runs of each model were executed in 8 threads using a time budget of 7200 seconds for available benchmark instances (Kletzander, Gjergji, and Musliu 2024). The instances for the AMRP-D are designed based on a prac-

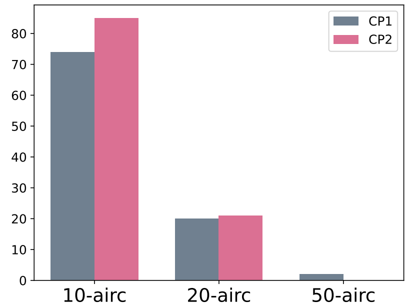

Figure 2: Number of solved instances based on instance size for `CP1` and `CP2`.

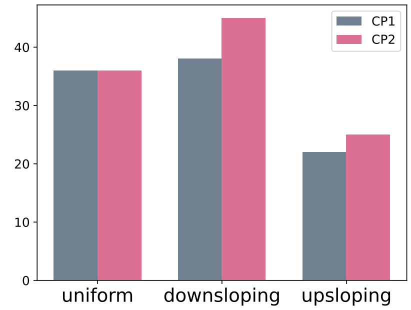

Figure 3: Number of solved instances based on instance structure for `CP1` and `CP2`.

tical application considering a single hub, flight legs with medium to long distance, daily departure peaks, and seasonal fluctuations. The scheduling period is 7, 14, or 28 days. Regarding the density of the flight legs to be assigned, instances have uniform density, downsloping density with demand that decreases in the scheduling period, or upsloping density with increasing number of flight legs during the scheduling period. The number of flight legs varies from 70 to 1695, the number of aircraft is 10, 20, or 50. Different combinations of these characteristics result in 324 instances altogether.

Feasibility Results

The model presented by Gjergji et al. (2025) is denoted as `CP1` and the model proposed in this paper is shown as `CP2`. Out of 324 instances in total, `CP1` solves 96 instances, while `CP2` solves 106 instances. In Figure 2, the results of both models are compared based on the number of aircraft. `CP2` performs better for instances with 10 aircraft. For instances with 20 aircraft the performance is similar (only 1 more instance solved by `CP2`). On the other hand, `CP2` does not solve any instance with 50 aircraft. The results of the two models depending on the density of the instances are displayed in Figure 3. For instances with uniform density both models have the same performance. For instances with downsloping and upsloping density `CP2` has a better performance. Runtime results for instances which at least one model did not time out are given in Figure 4, where `CP1` is shown on x-axis and `CP2` on the y-axis. `CP2` offers the lowest runtime for 127 instances, `CP1` is faster for 35 instances, and both models have the same runtime for only 1 instance. Further analysis on the statistics outputted by FlatZinc shows that both the number of variables and the number of constraints is much lower for `CP2` compared to `CP1`. Such findings support the results of `CP2` being faster on most instances, as well as solving more instances.

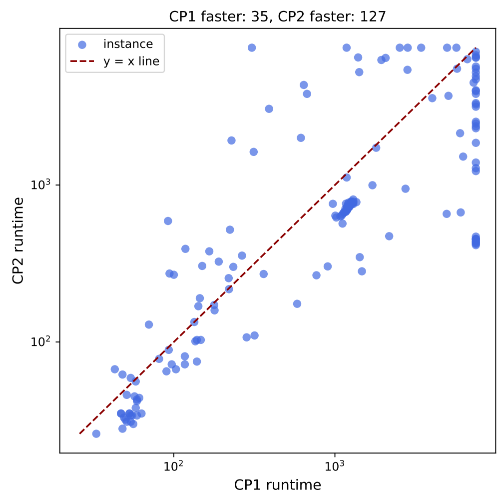

Figure 4: Comparison of `CP1` & `CP2` in terms of runtime. Logarithmic scale has been applied to the actual runtime values as they range from seconds to the timeout limit.

Discussion and Conclusion

In this paper we presented an alternative constraint programming model for the aircraft maintenance routing problem with a distribution objective. This study is motivated by the question whether providing a different way of modeling consecutive flight legs and substituting optional variables with regular variables could lead to better results. Specifically, the new model employs a regular variable for maintenance tasks and uses dummy variables to cover the maintenance tasks that are not required in the scheduling period. Furthermore, the assignment of the flight legs is done with a regular variable indicating the aircraft operating each leg, and an explicit variable for the previous leg. This results is a much lower number of variables and constraints in the flattened model, which ultimately contributes to a higher number of feasible solutions and faster runtime, in particular for smaller instances and changing demand density. Many aspects could be considered for further refining the model: reformulating the complex constraints in Equation (10) and (11), deriving a tighter bound for the maximum number of maintenance tasks mm in the scheduling period, and integrating the proposed CP model in the decomposition approach (Gjergji et al. 2025).

Acknowledgements

The financial support by the Austrian Federal Ministry of Labour and Economy, the National Foundation for Research, Technology and Development and the Christian Doppler Research Association is gratefully acknowledged. This research was also funded by the Austrian Science Fund (FWF), project I 5443-N, Grant-DOI 10.55776 and by the Doctoral Program Vienna Graduate School on Computational Optimization, Austrian Science Foundation (FWF), under grant No.: W1260-N35 (https://vgsco.univie.ac.at/). The work of P.J. Stuckey and H. Bierlee is partially supported by the Australian Research Council (OPTIMA ITTC IC200100009).

References

Eltoukhy, A. E.; Chan, F. T.; and Chung, S. H. 2017. Airline schedule planning: a review and future directions. *Industrial Management & Data Systems*, 117(6): 1201–1243.

Gjergji, I.; Kletzander, L.; Bierlee, H.; Musliu, N.; and Stuckey, P. 2025. Combining Constraint Programming and Metaheuristics for Aircraft Maintenance Routing with a Distribution Objective. In Tack, G., ed., *Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, 1–17. Cham: Springer Nature Switzerland. ISBN 978-3-031-95976-9.

Kletzander, L.; Gjergji, I.; and Musliu, N. 2024. Combining Aircraft Maintenance Routing with a Distribution Objective. In *Proceedings of the 14th International Conference on the Practice and Theory of Automated Timetabling, PATAT 2024*, 325–328.

Nethercote, N.; Stuckey, P. J.; Becket, R.; Brand, S.; Duck, G. J.; and Tack, G. 2007. MiniZinc: Towards a Standard CP Modelling Language. In Bessiere, C., ed., *Principles and Practice of Constraint Programming - CP 2007, 13th International Conference, CP 2007, Providence, RI, USA, September 23-27, 2007, Proceedings*, volume 4741 of *Lecture Notes in Computer Science*, 529–543. Springer.

Temucin, T.; Tuzkaya, G.; and Vayvay, O. 2021. Aircraft maintenance routing problem–a literature survey. *Promet-Traffic&Transportation*, 33(4): 491–503.

Xu, Y.; Wandelt, S.; and Sun, X. 2023. Airline Scheduling Optimization: Literature Review and a Discussion of Modeling Methodologies. *Intelligent Transportation Infrastructure*.