Scheduling Aircraft Landings

João Sousa[1] [0000−0003−0824−1303]

Faculdade de Engenharia da Universidade do Porto
R. Dr. Roberto Frias, 4200-465 Porto
joao.pedro.sousa@ieee.org

Abstract. This report addresses the Aircraft Landing Scheduling Problem (ALSP) in the context of multiple-runway scenarios. The objective is to determine feasible landing times for a given set of aircraft, ensuring that each plane lands within its designated time window while maintaining minimum separation requirements between consecutive landings. The problem is formulated using both Mixed-Integer Programming (MIP) and Constraint Programming (CP) approaches: the MIP model utilizes binary variables to capture the relative landing order of aircraft and continuous variables for the assignment of landing times, while the CP model leverages constraint satisfaction and logical reasoning to enforce the scheduling requirements.
To evaluate the performance and limitations of the proposed models, computational experiments are conducted on test instances of increasing size. The analysis begins with small-scale scenarios and incrementally increases the number of aircraft, allowing the identification of key parameters that influence the complexity of the problem. Both models are solved using CPLEX and the study examines solution quality, computational effort, and scalability.
This report presents the complete formulations for both MIP and CP approaches, discusses their structures, and provides initial results from the experimental analysis.

Keywords: Aircraft Landing Scheduling Problem · Mixed-Integer Programming · Constraint Programming · Optimization · Decision Support.

1 Introduction

1.1 Problem Description

The Aircraft Landing Scheduling Problem (ALSP) is a fundamental challenge in air traffic control and airport operations. It arises in situations where multiple aircraft must be scheduled to land on a single runway, typically within a constrained time horizon. Each aircraft has a specific time window—defined by its earliest and latest acceptable landing times—within which it must land. Additionally, a minimum separation time must be maintained between landings of consecutive aircraft to ensure safety, accounting for wake turbulence and runway occupancy constraints.

These separation times vary based on the characteristics of the aircraft involved, such as weight class or speed. For example, a heavy aircraft followed by a lighter one may require a longer separation interval. Violating these constraints may lead to unsafe conditions or operational inefficiencies. Furthermore, deviations from preferred landing times can incur costs, such as fuel consumption for circling or delays in turnaround schedules.

The ALSP, even in its single-runway form, is a complex optimization problem due to its combinatorial nature. It is classified as NP-hard, and exact solutions become increasingly difficult to obtain as the number of aircraft grows. For these reasons, it provides a rich and practical case study for applying advanced optimization and decision support techniques.

1.2 Objectives of the Report

The main objective of this report is to formulate and analyze mathematical optimization models for the ALSP using both MIP and CP approaches. This includes:

- Developing MIP and CP models that capture the essential constraints and objectives of the problem.
- Analyzing the impact of different instance parameters, such as the number of aircraft and the tightness of time windows, on the solvability of both models.
- Testing both models on instances of increasing complexity to evaluate their scalability and computational performance.
- Identifying practical limitations and proposing directions for improving solution efficiency, such as through pre-processing or parameter tuning.

2 The MIP Model

This section presents the Mixed Integer Programming (MIP) formulation for the Aircraft Landing Scheduling Problem (ALSP) on multiple runways, adapted from the work of Beasley et al. [1]. The objective is to determine optimal landing times and runway assignments for a set of aircraft so that operational requirements constraints are satisfied, while minimizing total penalties for landing earlier or later than target times.

Let P denote the number of aircraft to be scheduled, and R the number of available runways. For each aircraft $i \in \{1, \ldots, P\}$, the following input parameters are considered:

- A_i: Appearance time of aircraft i (the earliest time at which the aircraft is ready to land)
- E_i: Earliest feasible landing time for aircraft i
- L_i: Latest feasible landing time for aircraft i
- T_i: Target (preferred) landing time for aircraft i
- g_i: Penalty cost per unit time for landing earlier than T_i
- h_i: Penalty cost per unit time for landing later than T_i

- $S_{i,j}$: Required minimum separation time between aircraft i and j if i lands before j on the same runway

The main decision variables are:

- x_i: Actual landing time assigned to aircraft i
- α_i: Earliness of aircraft i, representing the amount of time landing before T_i
- β_i: Lateness of aircraft i, representing the amount of time landing after T_i
- $y_{i,j}$: Binary variable equal to 1 if aircraft i lands before aircraft j, 0 otherwise
- $\gamma_{i,r}$: Binary variable equal to 1 if aircraft i is assigned to runway r, 0 otherwise
- $\delta_{i,j}$: Binary variable equal to 1 if aircraft i and j are assigned to the same runway, 0 otherwise

2.1 Objective Function

The objective is to minimize the total penalty associated with earliness and lateness across all aircraft:

$$\min \sum_{i=1}^{P} (g_i \cdot \alpha_i + h_i \cdot \beta_i) \tag{1}$$

2.2 Constraints

The model is subject to the following constraints:

Time Window Constraints Each aircraft must land within its prescribed time window:

$$E_i \leq x_i \leq L_i \quad \forall i \in \{1, \dots, P\} \tag{2}$$

Runway Assignment Constraints Each aircraft must be assigned to exactly one runway:

$$\sum_{r=1}^{R} \gamma_{ir} = 1 \quad \forall i \in \{1, \dots, P\} \tag{3}$$

where γ_{ir} is 1 if aircraft i lands on runway r, and 0 otherwise.

Same Runway Indicator Constraints For each pair of distinct aircraft, define whether they share the same runway:

$$\delta_{ij} \geq \gamma_{ir} + \gamma_{jr} - 1 \quad \forall i, j \in \{1, \dots, P\},\, i \neq j,\, \forall r \in \{1, \dots, R\} \tag{4}$$

where $\delta_{ij} = 1$ if both aircraft are assigned to the same runway.

Separation Constraints For any pair of aircraft assigned to the same runway, enforce minimum separation between their landing times:

$$x_j - x_i \geq S_{ij} \cdot \delta_{ij} - M \cdot y_{ji} \quad \forall i, j \in \{1, \ldots, P\},\ i \neq j \tag{5}$$

where y_{ij} indicates whether i lands before j, and M is a sufficiently large constant.

Landing Order Exclusivity For each pair of aircraft, exactly one must be designated as landing before the other:

$$y_{ij} + y_{ji} = 1 \quad \forall i, j \in \{1, \ldots, P\},\ i \neq j \tag{6}$$

Landing Time Decomposition The relationship between actual landing time, target time, earliness, and lateness is expressed as:

$$x_i - T_i = \alpha_i - \beta_i \quad \forall i \in \{1, \ldots, P\} \tag{7}$$

Non-negativity All time and penalty variables must be non-negative:

$$x_i \geq 0, \quad \alpha_i \geq 0, \quad \beta_i \geq 0 \quad \forall i \in \{1, \ldots, P\} \tag{8}$$

This formulation comprehensively models the multi-runway aircraft landing scheduling problem using a mixed integer programming approach.

3 The Constraint Programming Model

This section describes the constraint programming (CP) formulation for the Aircraft Landing Scheduling Problem (ALSP) on multiple runways. The CP approach leverages native constraint satisfaction and logic reasoning capabilities to efficiently model scheduling and separation requirements without the need for binary or big-M constraints.

Let P be the number of aircraft and R the number of available runways. The key parameters are as defined in previous section and variables are defined as follows:

- x_i: Actual landing time assigned to aircraft i
- r_i: Runway assigned to aircraft i, taking integer values in $\{1, \ldots, R\}$
- α_i: Earliness of aircraft i, defined as $\max(0, T_i - x_i)$
- β_i: Lateness of aircraft i, defined as $\max(0, x_i - T_i)$

3.1 Objective Function

The objective is to minimize the total cost associated with early and late landings:

$$\min \sum_{i=1}^{P} (g_i \cdot \alpha_i + h_i \cdot \beta_i) \tag{9}$$

3.2 Constraints

The CP model enforces the following constraints:

Time Window Constraints Each aircraft must land within its allowable time window:

$$E_i \leq x_i \leq L_i \quad \forall i \in \{1, \ldots, P\} \tag{10}$$

Runway Assignment Constraints Each aircraft is assigned a runway indexed from 1 to R:

$$1 \leq r_i \leq R \quad \forall i \in \{1, \ldots, P\} \tag{11}$$

where r_i denotes the runway assigned to aircraft i.

Earliness and Lateness Constraints For each aircraft, earliness and lateness are defined as:

$$\alpha_i \geq T_i - x_i \quad \forall i \in \{1, \ldots, P\} \tag{12}$$

$$\beta_i \geq x_i - T_i \quad \forall i \in \{1, \ldots, P\} \tag{13}$$

Runway Separation Constraints For every pair of distinct aircraft, if both are assigned to the same runway, then the required separation time between their landings must be satisfied. In constraint programming, this condition can be written using logical disjunction:

$$\forall i, j \in \{1, \ldots, P\},\ i < j,\ \text{if } S_{ij} > 0: \tag{14}$$

$$(r_i \neq r_j) \ \vee \ (x_i + S_{ij} \leq x_j) \ \vee \ (x_j + S_{ji} \leq x_i)$$

That is, either the two aircraft are assigned to different runways, or the required separation time is respected in one direction or the other.

Non-negativity Earliness and lateness must be non-negative:

$$\alpha_i \geq 0, \quad \beta_i \geq 0 \quad \forall i \in \{1, \ldots, P\} \tag{15}$$

This formulation defines the constraint programming model for the multi-runway aircraft landing scheduling problem. The subsequent sections present computational experiments and discuss the observed results.

4 Computational Results and Instance Scaling

This section presents the preliminary computational results obtained by applying the proposed MIP model to a series of standard benchmark instances from the OR-Library dataset for the Aircraft Landing Scheduling Problem [2]. Specifically, we use instances `airland1` through `airland8`, all simulated from 1 to 4 runways. These instances are well-established in the literature and provide a suitable basis for testing both the validity and scalability of our model.

4.1 Experimental Setup

All computations were carried out on a machine equipped with an **Intel Core i7-1355U CPU** and **16GB of RAM**, using CPLEX solver. Each instance was run individually, with a time limit of 60 seconds imposed per instance to control computational effort.

4.2 Results on Benchmark Instances

Tables 1 and 2 summarize the solver performance across the benchmark instances. For each instance, the tables report the number of aircraft, the total earliness/lateness penalty in the optimal solution, and the runtime required to reach the solution.

Table 1. Results for OR-Library benchmark instances (`airland1`–`airland8`) using MILP.

Instance	# Aircraft	# Runways	# Binaries	# Solutions	Optimal?	Optimal Value	Total Time (s)
airland1	10	1	45	4	Yes	700	0.03
		2	145	4	Yes	90	0.03
		3	165	4	Yes	0	0.00
		4	175	4	Yes	0	0.02
airland2	15	1	105	4	Yes	1480	0.06
		2	330	6	Yes	210	0.03
		3	360	4	Yes	0	0.02
		4	375	4	Yes	0	0.04
airland3	20	1	190	4	Yes	820	0.06
		2	560	4	Yes	60	0.08
		3	630	4	Yes	0	0.03
		4	650	4	Yes	0	0.03
airland4	20	1	190	7	Yes	2520	0.06
		2	590	4	Yes	640	0.09
		3	630	4	Yes	130	0.14
		4	650	4	Yes	0	0.09
airland5	20	1	190	6	Yes	3100	0.14
		2	590	6	Yes	650	0.36
		3	630	8	Yes	170	0.2
		4	650	6	Yes	0	0.06
airland6	30	1	-	-	Yes	24442	0.0
		2	232	8	Yes	554	0.19
		3	292	7	Yes	0	0.05
		4	332	7	Yes	0	0.03
airland7	44	1	22	2	Yes	1550	0.05
		2	282	3	Yes	0	0.05
		3	370	5	Yes	0	0.05
		4	414	5	Yes	0	0.03
airland8	50	1	1207	6	Yes	1950	0.36
		2	3691	7	Yes	135	0.56
		3	3791	6	Yes	0	0.34
		4	3841	6	Yes	0	0.34

Table 2. Results for OR-Library benchmark instances (`airland1`–`airland8`) using CP.

Instance	# Aircraft	# Runways	Memory (MB)	Best Bound	Optimal?	Optimal Value	Total Time (s)
airland1	10	1	10.8	700	Yes	700	0.14
		2	8.9	90	Yes	90	0.06
		3	7.9	0	Yes	0	0.03
		4	7.6	0	Yes	0	0.03
airland2	15	1	18.1	1480	No	1480	7.32
		2	11.6	210	Yes	210	0.09
		3	9.1	0	Yes	0	0.03
		4	8.9	0	Yes	0	0.03
airland3	20	1	16.8	820	Yes	820	0.56
		2	13.4	60	Yes	60	0.09
		3	10.8	0	Yes	0	0.04
		4	11.3	0	Yes	0	0.04
airland4	20	1	24.5	210	Timeout	2520	60
		2	26.0	1	Timeout	640	60
		3	10.8	130	Yes	130	0.14
		4	11.3	0	Yes	0	0.09
airland5	20	1	24.3	241	Timeout	3100	60
		2	25.8	1	Timeout	650	60
		3	10.8	170	Yes	170	0.20
		4	11.3	0	Yes	0	0.06
airland6	30	1	26.8	24442	Yes	24442	0.77
		2	21.8	554	Yes	554	0.14
		3	17.8	208	Yes	208	0.13
		4	17.8	0	Yes	0	0.15
airland7	44	1	54.9	106	Timeout	1550	60
		2	17.2	970	Yes	970	0.04
		3	11.8	900	Yes	900	0.04
		4	11.8	820	Yes	820	0.05
airland8	50	1	43.3	210	Timeout	1950	60
		2	40.7	135	Yes	135	0.56
		3	27.2	0	Yes	0	0.34
		4	26.5	0	Yes	0	0.34

4.3 MILP results

The MILP model was tested on standard benchmark instances (`airland1 - airland8`) across increasing numbers of aircraft and runways. According to 1:

- Optimality and Solution Quality: For all instances and configurations, the MILP model consistently found optimal solutions. The optimal values for the total earliness/lateness penalties are reported, with several cases achieving a penalty of zero, especially when more runways are available.
- Scalability: Even as the number of aircraft increases from 10 to 50, and the number of runways from 1 to 4, the model maintains efficient performance. For instance, *airland8* (50 aircraft, 4 runways) still achieves an optimal solution with zero penalty.
- Computation Time: Solution times are extremely low for all problem sizes. Even the largest instance (airland8 with 50 aircraft and 4 runways) is solved in 0.34 seconds. This highlights the effectiveness of the MILP formulation and the efficiency of the solver, at least within the tested size range.
- Complexity Growth: The number of binary variables (linked to the ordering of aircraft) grows quickly with the number of aircraft and runways. Despite this, solution times remain low, which suggests the instances are not yet at the scale where MILP becomes infeasible.

4.4 CP results

The CP model results are summarized in 2:

- Optimality and Solution Quality: The CP model also achieves optimal solutions in most cases, matching the MILP results for both solution quality and objective value. However, for some larger instances or tighter configurations, it only reaches the best bound within the time limit.
- Scalability: The CP approach starts to struggle with some larger and/or tighter instances. For example, for *airland4* and *airland5* (20 aircraft, 1–2 runways), the solver hits the 60-second time limit and returns the best bound found so far.
- Computation Time and Memory Usage: For small and medium instances, solution times are slightly higher than MILP (e.g., airland1, 10 aircraft, takes 0.14 seconds for CP versus 0.03 seconds for MILP). For larger instances, particularly with tight separation and time windows, computation times can increase sharply, and timeouts occur. Memory usage grows as well, but remains moderate (generally under 55 MB).
- Timeouts: CP hits timeouts for larger and tighter instances, especially for 1–2 runway scenarios.

5 Conclusions

The computational study of the Aircraft Landing Scheduling Problem using both Mixed-Integer Linear Programming and Constraint Programming models reveals distinct performance profiles for each approach:

- MILP demonstrates superior scalability and computational efficiency, consistently solving all benchmark instances—including those with up to 50 aircraft and multiple runways—in under a second. The robustness of MILP across varying problem sizes and constraints highlights its suitability for large-scale or tightly constrained ALSP scenarios.
- CP, while competitive in terms of solution quality for less complex instances, becomes increasingly sensitive to problem hardness as the number of aircraft increases or as scheduling constraints become tighter. Although CP often finds optimal or near-optimal solutions, it is more prone to timeouts and exhibits longer computation times in challenging instances. Memory usage is slightly higher for CP, but does not present a limiting factor within the tested range.
- The choice between the two should thus be guided by the specific characteristics of the scheduling task: MILP excels for larger, resource-constrained, or industrially relevant instances, while CP may be advantageous for problems emphasizing complex logical or custom constraints that are more naturally expressed within the CP paradigm.

In summary, for the ALSP as presented here, MILP provides a more robust and efficient framework, especially as the scale or tightness of the scheduling problem increases. CP remains a valuable tool for specialized use cases, but MILP should be the default method of choice for general-purpose, scalable aircraft landing scheduling.

References

1. Beasley, J.E., Krishnamoorthy, M., Sharaiha, Y.M., Abramson, D.: Scheduling aircraft landings—the static case **34**(2), 180–197. https://doi.org/10.1287/trsc.34.2.180.12302, https://pubsonline.informs.org/doi/10.1287/trsc.34.2.180.12302
2. Beasley, J.: Or-library: Aircraft landing problem instances. https://people.brunel.ac.uk/ mastjjb/jeb/orlib/airlandinfo.html (2000), accessed: 2025-04-21