ISSN 1648-7788 / eISSN 1822-4180

AVIATION

2025
Volume 29
Issue 3
Pages 164–173
https://doi.org/10.3846/aviation.2025.24535

OPTIMIZING AIRCRAFT MAINTENANCE TASKS ALLOCATION USING MIXED INTEGER LINEAR PROGRAMMING

Karim ALI ✉, Darius RUDINSKAS, Virginija LEONAVIČIŪTĖ

Department of Aeronautical Engineering, Antanas Gustaitis' Aviation Institute, Vilnius Gediminas Technical University, Saulėtekio al. 11, LT-10223 Vilnius, Lithuania

Article History:
- received 5 May 2025
- accepted 18 June 2025

Abstract. This study addresses the optimization of aircraft maintenance task allocation for small fleets using Mixed-Integer Linear Programming (MILP). The research integrates manpower efficiency, regulatory compliance, and workload balancing to minimize downtime and enhance resource utilization. A mathematical model is formulated to account for task durations, skill levels, and sequential/parallel task constraints, validated via MATLAB implementation. Results from a simulated 50-hour Cessna 172 maintenance check demonstrate a 25% reduction in completion time compared to average manual scheduling. The model balances workloads by assigning tasks based on manpower expertise, highlighting the critical role of human factors in reducing errors and improving efficiency. Practical implications include a cost-effective alternative to commercial software for small operators, enabling optimized planning without high-cost tools. This work bridges a gap in maintenance literature by explicitly incorporating manpower efficiency into MILP frameworks, offering actionable insights for regulators and operators to avoid maintenance congestions and enhance operational resilience.

Keywords: aircraft maintenance, mixed-integer linear programming (MILP), manpower efficiency, optimization, task allocation.

✉Corresponding author. E-mail: *e.karimali320@gmail.com*

1. Introduction

Aircraft maintenance planning is a critical pillar of aviation safety, ensuring compliance with stringent regulations such as ICAO Annex 8 and EASA Part M while maintaining operational efficiency (European Union Aviation Safety Agency [EASA], n.d.-a). For small fleets such as flight training schools, agricultural aviation, or emergency services maintenance downtime directly impacts operational viability, as a single grounded aircraft can disrupt operation activities and cause maintenance congestion (Hasancebi et al., 2023). Despite advancements in optimization models for large-scale operations, existing frameworks often neglect the unique challenges of small fleets, including limited skilled manpower, high downtime costs, and reliance on costly commercial software (e.g., AMOS, TRAX). Prior studies on maintenance planning have focused on large fleets, assuming homogeneous manpower efficiencies and overlooking skill-level disparities. For example, Kozanidis et al. (2012), Sanchez et al. (2020) developed optimization models that minimize downtime but fail to account for manpower heterogeneity, a critical factor in real-world efficiency. Additionally, heuristic approaches such as Witteman et al. (2021) bin-packing algorithm prioritize computational speed over optimality, risking suboptimal resource allocation. Small fleets lack tailored solutions that balance manpower efficiency, regulatory compliance, and cost-effectiveness.

This study focuses on optimizing aircraft maintenance planning for small fleets by integrating manpower efficiency (e.g. 75% vs. 50% skill levels) into a Mixed-Integer Linear Programming (MILP) framework. To develop a MILP model that minimizes maintenance downtime while ensuring regulatory compliance and maximizing manpower utilization in small-fleet operations.

Research tasks:

1. Review existing optimization models and identify gaps in manpower efficiency integration.
2. Formulate a MILP model incorporating skill-based task allocation and safety constraints.
3. Validate the model using MATLAB, offering a cost-effective alternative to proprietary software.
4. Analyze case studies (e.g., Cessna 172 maintenance checks) to assess reductions in downtime and workload imbalance.

This bridges a critical gap by explicitly modeling manpower skill levels and regulatory constraints in MILP, a departure from prior studies that treat workforce as homogeneous. By leveraging MATLAB's Branch-and-Bound algorithm, the model provides an accessible solution for small operators, reducing reliance on expensive tools.

2. Literature review

In this review, the emphasis is placed on two primary aspects: first, to clarify the factors that have been employed in the optimization process and the construction of the optimization model, as example variables such as manpower quantity and feasibility; furthermore, it is imperative to comprehend how these studies have neglected the efficiency of human resources. Second, the review will concentrate on the optimization models that have been implemented and their subsequent applicability and examine the potential applications of these models as well as the intricacies inherent in their structure.

Existing research on aircraft maintenance optimization has overlooked the critical role of manpower efficiency and workforce heterogeneity, focusing instead on technical and logistical constraints (Table 1).

It is evident, that homogeneous manpower assumptions approach is effective for expedited scheduling, it poses a risk of suboptimal resource utilization in smaller fleets, where the heterogeneity of the manpower considerably influences maintenance checks accomplishment durations results.

Neglect of Human Factors: while Muecklich et al. (2023) linked 40% of ground-operation incidents to human error (e.g., procedural non-compliance, low adequate experience), most optimization studies such as Jordan and Azarm (2022), Madeira et al. (2021) ignored human factors in task allocation. Recent efforts in aviation safety have emphasized the importance of proactive hazard reporting, situational awareness, and resource management as part of a broader Safety Management System (SMS) framework (EASA, n.d.-b). A robust SMS explicitly integrates human performance metrics, such as skill-level variability and error probability, into maintenance workflows, yet these considerations remain absent in quantitative optimization models This disconnect highlights the need for an integrative approach that captures the interplay between operational efficiency and human performance constraints.

Prior studies employed diverse optimization techniques, each with distinct applicability, scale, and complexity. A critical evaluation of approaches used in scientific literature highlights limitations for resource-constrained operators (Table 2).

Multi-Objective Mixed-Integer Linear Programming (MMILP): extends MILP by simultaneously addressing conflicting objectives (e.g., cost minimization vs. resource availability). For example, Sanchez et al. (2020) used MMILP to optimize tail assignments and hangar scheduling for large fleets, achieving near-optimal solutions in minutes. However, its computational intensity requires iterative algorithms and large datasets renders it impractical for small fleets with limited IT infrastructure and fewer resources to manage complex trade-offs. Mixed-Binary Integer Nonlinear Programming (MBINLP): handles nonlinear relationships (e.g., variable maintenance intervals during disruptions). Jordan and Azarm (2022) applied MBINLP, adapting to flight cancellations and workforce shortages. However, its nonlinear complexity and reliance on machine learning for uncertainty modelling make it inaccessible for small operators lacking advanced tools. Second-Order Cone Programming (SOCP) Duran et al. (2014) used SOCP to optimize cruise times and reduce idle costs by 60%. While effective for large airlines. SOCP is a type of convex optimization problem that involves optimizing a linear objective function over the intersection of an affine set and the product of second-order cones, making it suitable for problems with conic constraints. In contrast, MILP involves linear objective functions and constraints, with some variables constrained to be integers, making it ideal for problems requiring discrete decisions. The following sections delve into the key differences and applications of SOCP and MILP. Heuristic and Metaheuristic Approaches: Heuristics like bin-packing algorithm did not achieve optimal solutions (Witteman et al., 2021) or genetic algorithms (Niu et al., 2021) prioritize computational speed over optimality. For instance, Witteman et al. (2021) reduced task allocation time by 30% using a constructive heuristic but acknowledged an optimality gap of up to 5%. Similarly, metaheuristics (e.g., bacterial foraging optimization) (Ribagin & Lyubenova, 2021) they usually need precise adjustment of parameters, which can take a lot of time and might

Table 1. Comparison between elements used in different research papers (source: compiled by the authors)

Reference	Approach	Findings / Results	Limitations
(Dinis & Barbosa-Póvoa, 2015)	Optimization framework for maintenance planning	Not explicitly stated (implied focus on technical/logistical constraints)	Assumed homogeneous manpower skill levels; ignored workforce heterogeneity
(Johny Ali Firdaus et al., 2020)	Linear programming for heavy maintenance centers	Achieved 90% resource utilization	Failed to account for skill-level disparities (e.g., 75% vs. 50% efficiency) or dynamic task assignments
Witteman et al. (2021)	Bin-packing heuristic for task allocation	30% reduction in task allocation duration; 5% optimality gap	Treated manpower as uniform; disregarded skill-level variations, leading to suboptimal resource utilization
Jordan and Azarm (2022)	Pandemic-era scheduling model	Adapted to manpower shortages	Overlooked efficiency losses due to skill mismatches; ignored human performance metrics

Table 2. Comparison between different optimization methods in different research papers (source: compiled by the authors)

Technique/ Method	Description	Key Benefits	Limitations for Small Operators	References
MMILP	Multi-Objective Mixed-Integer Linear Programming addressing conflicting objectives (e.g., cost vs. resource availability).	Achieves near-optimal schedules rapidly for large fleets (e.g., 529 aircraft).	Computationally intensive; requires large datasets and IT infrastructure.	Sanchez et al. (2020)
MBINLP	Mixed-Binary Integer Nonlinear Programming for nonlinear relationships (e.g., variable maintenance intervals).	Adapts to disruptions (flight cancellations, workforce shortages).	Complexity and reliance on machine learning for uncertainty modeling; inaccessible without advanced tools.	Jordan and Azarm (2022)
SOCP	Second-Order Cone Programming for convex optimization with conic constraints.	Reduces idle costs by 60% in large-scale airline scheduling.	Requires large-scale operations to justify setup; unsuitable for small fleets.	Duran et al. (2014)
Heuristics	Rule-based methods (e.g., bin-packing) for rapid task allocation.	30% faster scheduling with <5% optimality gap.	Suboptimal solutions; sacrifices precision for speed.	Witteman et al. (2021)
Metaheuristics	High-level strategies (e.g., genetic algorithms, bacterial foraging optimization).	Prioritizes computational speed and adaptability.	Time-consuming parameter tuning; requires expertise to avoid suboptimal performance.	Niu et al. (2021); Ribagin and Lyubenova (2021)
Reinforcement Learning (RL)	Dynamic schedule adjustments using real-time data.	Balances flight hours and maintenance intervals with a 0.12% optimality gap.	Relies on extensive training data; unsuitable for deterministic small fleets.	Guo and Wang (2023)
FMILP	Fuzzy Mixed-Integer Linear Programming for handling ambiguous constraints.	Reduces downtime in maintenance task scheduling.	Complexity from fuzzy logic; requires specialized knowledge.	Hasancebi et al. (2023)
ILP	Integer Linear Programming for optimizing utility capacity.	Enhances compliance in labor-intensive environments.	Limited flexibility for mixed-variable problems.	Hasancebi et al. (2023)
NLP	Nonlinear Programming for modeling system interactions.	Captures complex variable relationships.	Prone to local optima; convergence challenges.	(Sriram & Haghani, 2003)

need specialized knowledge (Ribagin & Lyubenova, 2021). This tuning process can lead to suboptimal performance if not done correctly. Reinforcement Learning (RL) dynamically adjusts schedules based on real-time data. Guo and Wang (2023) applied RL to balance flight hours and maintenance intervals. RL's reliance on extensive training data and stochastic environments makes it unsuitable for small fleets with deterministic schedules and limited historical data.

MILP (Mixed-Integer Linear Programming) is a highly adaptable and cost-effective method that allows for efficient resource allocation by incorporating specific constraints and objectives, making it particularly beneficial for small fleets. Its flexibility enables the integration of discrete variables, such as varying skill levels of maintenance personnel, alongside linear constraints that align with regulatory requirements, which is crucial given the manpower heterogeneity and compliance needs of small operators. Additionally, MILP can be solved using accessible and affordable tools like MATLAB, contrasting with more expensive commercial software options (Derbez & Lambin, 2022), thus offering a cost-effective option for operators with limited budgets. Its scalability allows it to effectively manage small-scale problems with fewer variables, unlike more complex approaches that require extensive computational resources. The primary advantage of MILP lies in its ability to integrate manpower efficiency and safety constraints, specifically addressing the unique challenges faced by small fleets while ensuring that maintenance planning is both efficient and compliant, all while remaining computationally lightweight (Schulze Spüntrup et al., 2021). This makes MILP an ideal choice for small operators seeking to optimize their operations while avoiding significant costs, ultimately enhancing productivity and maintaining regulatory compliance.

3. Methodology

The MILP approach is implemented using mathematical models in MATLAB, specifically using Branch and Bound (B&B) function which applied through MATLAB function (intlinprog) (Messac, 2015). B&B is an algorithmic technique that systematically explores all possible solutions to find the optimal one, by partitioning the problem into smaller subproblems (branching) and calculating bounds (bounding) to eliminate those subproblems that cannot yield a better solution than the current best (Figure 1). This method is particularly effective for solving combinatorial and discrete optimization problems, such as the travelling salesman problem and integer programming. Linear programming (LP), also known as linear optimization, is a method used to achieve the best possible outcome (such as maximum or minimum value) in a mathematical model characterized by linear relationships. It is a specialized form of mathematical programming that finds extensive application in various fields, including resource allocation, production planning, and scheduling (Cassel, 2021; Jordan & Azarm, 2022). Additionally, MILP extends linear programming by allowing some decision variables to be integers, providing greater modeling flexibility in real-world scenarios. MILP is particularly useful when discrete decisions are required, such as task assignments or equipment selection (Cassel, 2021; Qin et al., 2020). It consists of three key components of MILP:

1. An objective function that represents the goal of the optimization (such as minimizing cost, maximizing aircraft availability, or improving maintenance efficiency).
2. Decision variables (both continuous and integer), such as manpower allocation, specific maintenance tasks to be performed, and available time slots.
3. Constraints that represent the limitations and requirements of the problem, such as regulations regarding mandatory maintenance intervals, double inspections, required inspection item tasks, or human factors like crew fatigue limitations and efficiency.

The main MILP formula that the suggested model use is defined as:

$$f\left(x_1, x_2, \cdots, x_n\right) = c_1x_1 + c_2x_2 + \cdots + c_nx_n = \sum_{i=1}^{n} c_ix_i; \quad (1)$$

$$x_i \in \mathbb{Z} \ \forall i; \quad (2)$$

$$c = \begin{bmatrix} c_1 \\ c_2 \\ c_n \end{bmatrix} \quad \text{and} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_n \end{bmatrix}; \quad (3)$$

$$c^T x = \left[c_1 c_2 \cdots c_n\right] \begin{bmatrix} x_1 \\ x_2 \\ x_n \end{bmatrix} = c_1x_1 + c_2x_2 + \cdots + c_nx_n, \quad (4)$$

where: c is vector of coefficients, x is vector of decision variables (integer variables), n represents number of elements and i is the index of element.

The linear inequality constraints are defined by feasible region or limits for the decision variables (x), as follows:

$$Ax \le b, \quad (5)$$

where: A is matrix of coefficients in for the linear inequality constraints, b is vectors of constants responding to the upper bounds. The equality constraints are defined as follows:

$$A_{eq}x = b_{eq}, \quad (6)$$

where: A_{eq} is matrix coefficient for the equality constraints, b_{eq} is vector of constants representing the exact values the linear combination must equal. The nonnegative constraints are set as

$$x_i \ge 0. \quad (7)$$

The relationship between maintenance checks and manpower is established seeking to optimize operations under constraints, such as limited manpower or time. Since the function f(x) to be optimized is the check's elapsed time, which is minimized by optimal task allocation, consequently, the check completion time must be defined and formalized as follows:

$$f\left(x_1, x_2, \cdots, x_n\right) = \varepsilon_c. \quad (8)$$

The duration of the maintenance check is denoted by ε_c, representing the total time for completion. Since a maintenance check consists of multiple tasks, the duration of each task is denoted as ε_t (in hours), indicating the time required for its completion. Task duration is measured in man-hours, which represents the total time needed for one person to complete the task. Certain tasks can be performed independently and in parallel, whereas others require a sequential approach for completion.

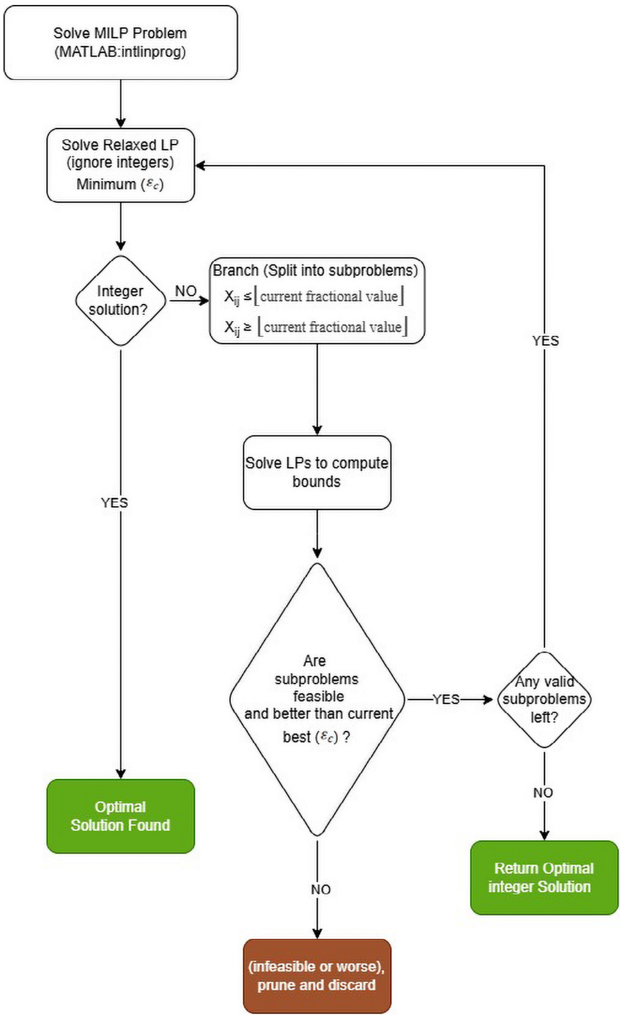

Figure 1. Branch & Bound by MATLAB (Intlinprog) function (source: compiled by the authors)

3.1. Parallel, sequential tasks and total check duration

The parallel tasks are executed within the same time frame as Figure 2 performing multiple maintenance tasks at the same time by different technicians. For example, one technician inspects the engine while another checks the landing gear simultaneously. Consequently, the total elapsed duration for the entire maintenance check must exceed the maximum elapsed duration of the individual tasks.

$$\varepsilon_{cp} \geq \max_i \left(\frac{\varepsilon_{ti}}{M_i} \right), \quad (9)$$

where: ε_{cp} is check elapsed duration for all tasks performed in parallel in (hours), $\{\phi\}_i$ is task elapsed duration for task (i) in Man-hour, M_i is amount of manpower for each task (i), i is task index number (in parallel method).

For the scenario where all tasks are performed in a sequence as Figure 2, Sequential task execution involves performing maintenance tasks one after another by the same or different technicians. For example, an inspection must be completed before a component can be reinstalled. total check duration is given by the following Equation:

$$\varepsilon_{cs} \geq \sum_{0}^{n} \left(\frac{\varepsilon_{tn}}{M_n} \right), \quad (10)$$

where: ε_{cs} is check total duration for all tasks performed sequentially, ε_{tn} is task elapsed duration for task (n) in Man-hour, M_n is amount of manpower for each task (n), n is index of the task number (in the sequential method).

The comparison between parallel and sequential task execution highlights significant differences in their impact on total maintenance check duration (Figure 2). Parallel execution allows multiple technicians to work simultaneously on different tasks, which can greatly reduce overall elapsed time. However, its effectiveness depends on the independence of tasks, manpower availability, and workspace constraints. In contrast, sequential execution increases the total check duration since each task must wait for the completion of the previous one, making it less efficient in terms of time utilization. Sequential scheduling is typically required when tasks are interdependent, limited manpower, when physical space or tooling limits simultaneous operations, or when procedural or safety requirements enforce a specific order.

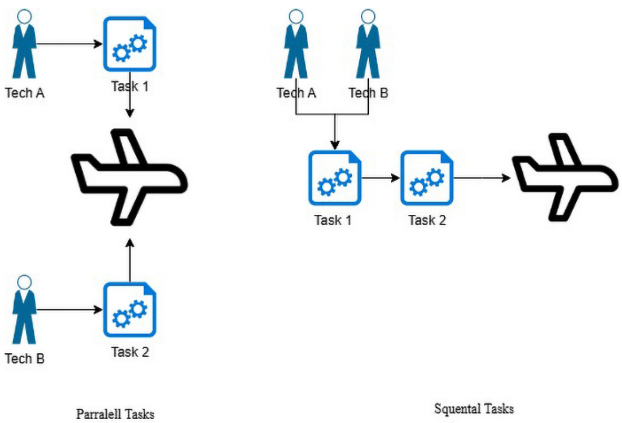

Figure 2. Comparison between parallel and sequential task (source: compiled by the authors)

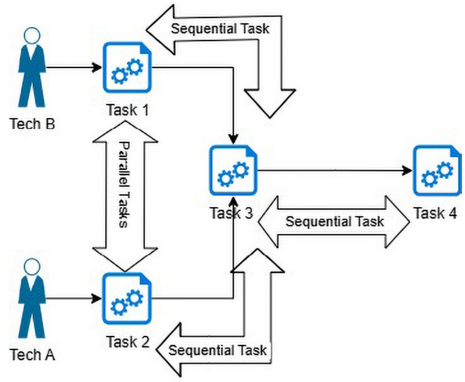

Figure 3. Combination of parallel and sequential tasks (source: compiled by the authors)

In the case of both task modes are combined, the total check elapsed time is represented as follows:

$$\varepsilon_c \geq \varepsilon_{cp} + \varepsilon_{cs}. \quad (11)$$

$$\varepsilon_c \geq \max_i \left(\frac{\varepsilon_{ti}}{M_i} \right) + \sum_{0}^{n} \left(\frac{\varepsilon_{tn}}{M_n} \right). \quad (12)$$

In practice a combined tasks allocation strategy, integrating both parallel and sequential task flows more accurately reflects typical maintenance practice (Figure 3). It optimizes technician efficiency, reduces idle time, and ensures procedural correctness.

3.2. Variables and constraints

Two types of variables are employed in the proposed model: binary decision variables, which indicate task assignments, and continuous variables, which track manpower completion times.

The binary variable (task assignment variable) x_{ij} indicates whether task (i) is assigned to manpower (j).

$$x_{ij} \begin{cases} (1)\, if\, the\, task\, (i)\, assigned\, to\, manpower\, (J) \\ (0)\quad otherwise \end{cases}.$$

In scenario I, $i \in \{1,2,3\}$ .corresponds to the task index, and $j \in \{1,2\}$ corresponds to the manpower index. These variables represent the core decision of the problem namely, which tasks are assigned to each technician. The manpower completion time variable T_j represents the total time manpower (technician) (j) takes to complete the assigned tasks, adjusted for efficiency factor e_j. Formally:

$$T_j = \sum_{i=1}^{3} \left(\frac{\varepsilon_{ti}}{e_j} \cdot x_{ij} \right), \quad (13)$$

where ε_{ti} is the duration of task i. These variables represent how long each technician remains occupied based

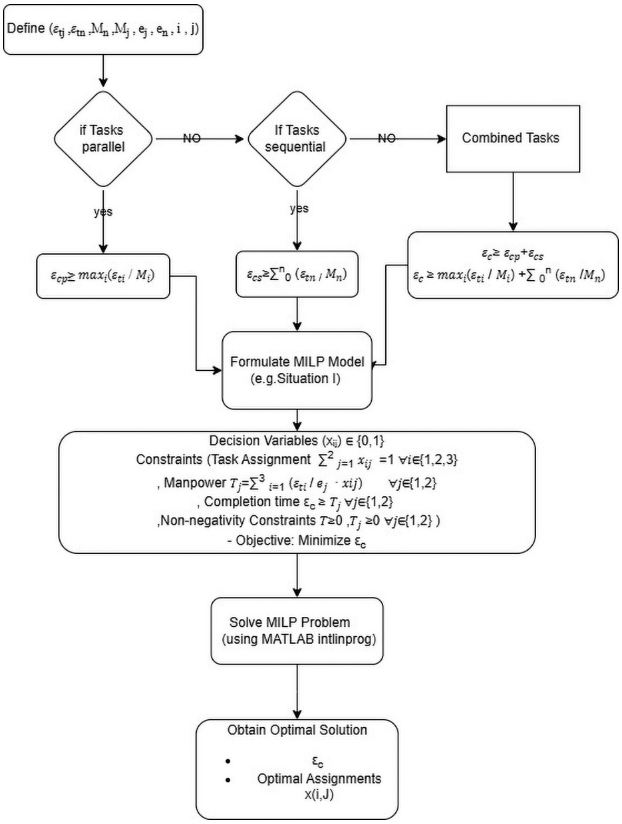

Figure 4. The proposed MILP-based maintenance allocation model (source: compiled by the authors)

on their assignments. The total completion time T, is a continuous variable, representing the maximum completion time *Tj* among all technicians:

$$T = \max_j \{T_j\}. \quad (14)$$

Finally, the overall check duration ε_c must be at least as large as *T*, ensuring that no tasks extend beyond the total allocated time:

$$\therefore \varepsilon_c \geq T. \quad (15)$$

To ensure the feasibility, efficiency, and logical consistency of the proposed maintenance check model, a set of mathematical constraints is defined. These constraints govern the behavior of task assignments, manpower workload, and the total check duration within the MILP formulation (Figure 4). Each constraint serves a specific operational purpose ranging from enforcing one-to-one task-to-technician allocation to limiting technician working hours based on efficiency. The constraints also reflect practical maintenance planning realities, such as maximum allowable check time and non-negativity of duration variables.

The constraints that are used in the model (Figure 4) seeking to ensure a realistic and solvable optimization problem are presented below, grouped according to their functional roles within the model, highlighting their role in shaping a realistic and solvable optimization model.

Assignment constraints – each task must be assigned exactly to one manpower:

$$\sum_{j=1}^{2} x_{ij} = 1 \quad \forall_i \in \{1,2,3\}. \quad (16)$$

Manpower time constraints – computes the time each worker spends on their assigned tasks, accounting for efficiency:

$$T_j = \sum_{i=1}^{3} \left(\frac{\varepsilon_{ti}}{e_j} \cdot x_{ij}\right) \quad \forall_j \in \{1,2\}. \quad (17)$$

Total completion time constraint – ensuring T is the maximum of T_j:

$$T \geq T_j \quad \forall_j \in \{1,2\}. \quad (18)$$

Non-negativity constraints – time cannot be negative:

$$T \geq 0, T_j \geq 0 \quad \forall_j \in \{1,2\}. \quad (19)$$

4. Results

By applying the suggested optimization method to the scenario described in Table 1, all feasible task assignments for Technicians A and B can be systematically evaluated. The goal is to minimize the overall check elapsed time by determining the optimal distribution of tasks between the two technicians. By applying the method (Equations (13) and (16)) on the scenario:

$$T_1 = \frac{1}{0.75}\left(x_{11} + 0.75x_{21} + 0.5x_{31}\right); \quad (20)$$

$$T_2 = \frac{1}{0.5}\left(x_{12} + 0.75x_{22} + 0.5x_{32}\right). \quad (21)$$

Since x_{ij} must be binary variable, and with respect to the constrains (Equations (15), (17), (18)) and all possible assignments are generated by substituting x_{ij} by {0,1}. The resulting outcome solutions and scenarios are presented in Table 3, revealing that scenario 2 yields the shortest check elapsed time and all optimal scenarios are shown in bold fields.

Table 3. A summary of task assignment to technicians A and B with all feasible scenarios (source: compiled by the authors)

Scenario	Tech-A (tasks)	Tech-B (tasks)	T_1 (hrs)	T_2 (hrs)	T (hrs)
1	1,2,3	none	3	0.0	3
2	1,2	3	2	1.5	2
3	1,3	2	2.33	1	2.33
4	1	2,3	1.33	2.5	2.5
5	2,3	1	1.66	2	2
6	2	1,3	0.667	3.5	3.5
7	3	1,2	1	3	3
8	none	1,2,3	0.0	4.5	4.5

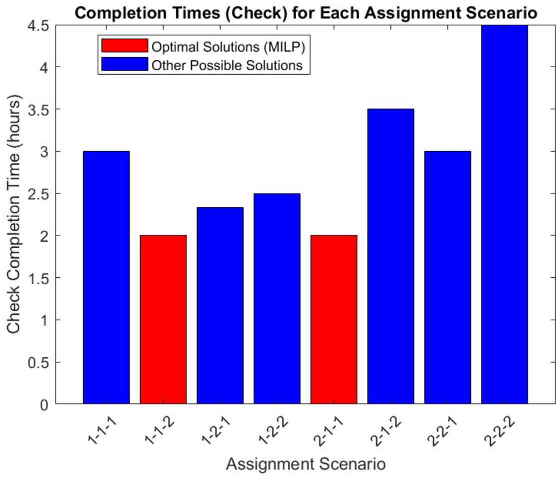

Figure 5. Completion times (Check) resulting from possible combinations of task assignments (source: compiled by the authors)

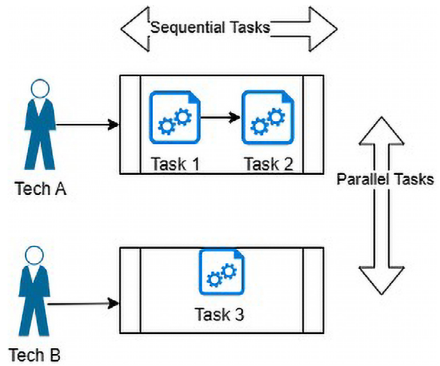

Figure 6. Scenario 2 with best task distribution (source: compiled by the authors)

The best scenarios (2nd and 5th columns) are shown in red while the other scenarios in blue (Figure 5). The obtained results show that, the combination of parallel and sequential tasks distribution (Figure 6) allows reduction of check completion time by 1 hour.

Brute-force enumeration was used to systematically evaluate all possible task assignment scenarios (e.g., 2 workers and 3 tasks = 8 total assignments) and compute their completion times. This serves as a verification step to confirm that the MILP solution truly achieves the optimal time (by comparing it against all feasible solutions).

4.1. Optimizing manpower vs workload: scenario I

To illustrate how different workload distributions affect the total completion time, two approaches to assigning tasks – parallel and sequential – were analysed. The scenario I defines available manpower (n = 2) with different efficiency levels: Technician A with efficiency 75% and Technician B with efficiency 50%, performing 3 tasks requiring [1, 0.75, 0.5] man-hour with different accomplishment times (Table 4).

Table 4. Example data (source: compiled by the authors)

Task no	Task Duration (mhrs)	Manpower	Manpower efficiency
1	1	Tech A	75%
2	0.75	Tech B	50%
3	0.5		

The workflow is structured into two dedicated work packages (A and B), each assigned to a specific technician (A or B). These work packages are executed in parallel (both technicians work simultaneously on their respective packages), while tasks within each package are performed sequentially (one after another by the assigned technician). The script used to determine the optimal task allocation – based on Equation (8) for minimizing total completion time. To ensure proper formulation for MILP, the model explicitly defines constraints (e.g., task assignment rules, manpower efficiency adjustments) and decision variables (both discrete xij for task assignments and continuous Tj for worker-specific completion times).

4.2. Optimizing manpower vs workload: scenario II – 50 hours check of Cessna 172 aircraft

To test the performance of the model in realistic set-up, a 50-hour check of Cessna 172 aircraft is considered, with the goal of minimizing the total completion time by optimally distributing tasks. The check includes 13 tasks, each taking between 0.5 and 1 hour (Table 5). Since the manufacturer does not provide specific task durations for the Cessna 172 (CESSNA 172S maintenance programme, 2022), these values are synthetic and randomly generated to reflect real-world variability. This choice does not affect the model's ability to determine the optimal solution, as the optimization process is driven by the structural relationships between tasks and technician efficiency, rather than the numerical value of specific task durations. In practice, substituting real-world task durations would yield different numerical results (e.g., total check completion time), but would not alter the model's structure, logic, or accuracy. Thus, the use of synthetic data in simulation affects only the numerical outcome, not the performance or applicability of the proposed optimization model.

Two technicians, technician A with efficiency of 75% and technician B with efficiency of 50% – are available for the check-up. To minimize total check duration, the workflow is divided into two work packages. Each package contains a group of tasks selected to achieve the shortest possible execution time and is assigned to one technician. Tasks within each package are performed sequentially, while both packages are executed in parallel. This configuration demonstrates the model's ability to balance manpower efficiency and task dependencies through a combined execution strategy, where sequential task execution within packages is complemented by parallel execution between packages.

Table 5. Time necessary to accomplish tasks (source: compiled by the authors)

Task No	1	2	3	4	5	6	7	8	9	10	11	12	13
Duration mhr	0.92	0.75	0.71	0.67	0.50	0.62	0.83	0.88	0.58	1.0	0.79	0.54	0.96

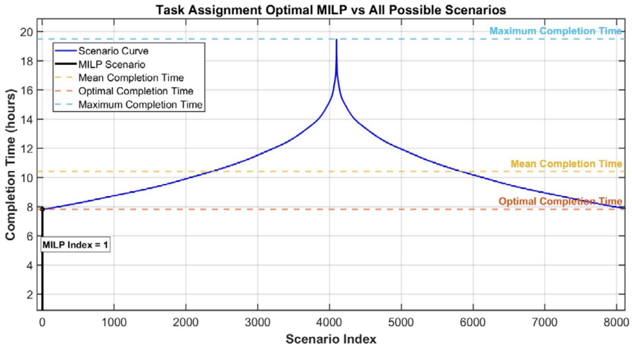

Figure 7. The optimal scenario vs all other scenarios (source: compiled by the authors)

Again Brute-force enumeration was used to systematically evaluate all possible task assignment scenarios (e.g., 2 Technicians and 13 tasks more than 8000 possible scenarios) and compute their completion times. This serves as a verification step to confirm that the MILP solution truly achieves the optimal time (by comparing it against all feasible solutions).

Optimal scenario for task distributions is summarized in Table 6, with total completion time of 7.82 hours, and the average completion time of 10.49 hours see Figure 7.

The obtained result reflects a 25.5% improvement over average scenarios, highlighting the model's ability to significantly reduce downtime. The optimization process intelligently favors assigning shorter tasks to the less efficient technician, maximizing parallel execution without causing delays. Technician A (75% efficiency) is assigned a larger number of tasks but completes them faster due to higher throughput. Technician B (50% efficiency) receives fewer but strategically selected tasks that fit his slower rate without creating bottlenecks.

To evaluate the effectiveness of the proposed MILP optimization model, statistical validation was performed by generating ten sets of random task durations see Table 7. and comparing total check completion times under both manual (random assignment) and MILP optimized task allocation strategies.

The average check duration for the manual scenarios was 10.54 hours, while the MILP-optimized scenarios achieved an average of 7.91 hours, representing an average improvement of 25%. A paired t-test confirmed that this reduction was statistically significant (p-value = 0.0000, $p < 0.05$), indicating that the MILP-based model provides a consistently superior solution compared to manual planning under varied task distributions.

5. Discussion and conclusions

The results of the study highlight several key findings relevant to the optimization of aircraft maintenance processes. In line with previous works, the MILP model demonstrated the benefits of optimal task allocation and manpower distribution to reduce maintenance time while adhering

Table 6. A summary of task assignments to each of two technicians for an optimal completion time (source: compiled by the authors)

Task No	1	2	3	4	5	6	7	8	9	10	11	12	13
Tech No	B	A	A	A	A	B	B	A	B	A	A	A	B

Table 7. Comparison of MILP and manual check times using random task durations (source: compiled by the authors)

NO	MILP times(hr)	Mean times(hr)	Task durations assumed (mhr)
1	8.54	11.39	0.91 0.95 0.56 0.96 0.82 0.55 0.64 0.77 0.98 0.98 0.58 0.99 0.98
2	8.58	11.43	0.74 0.90 0.57 0.71 0.96 0.90 0.98 0.83 0.52 0.92 0.97 0.84 0.88
3	7.58	10.11	0.87 0.70 0.83 0.59 0.85 0.52 0.64 0.52 0.55 0.91 0.85 0.66 0.98
4	7.84	10.44	0.52 0.72 0.69 0.88 0.90 0.59 0.74 0.72 0.82 0.85 0.88 0.64 0.84
5	7.87	10.48	0.83 0.58 0.56 0.75 0.98 0.67 0.79 0.61 0.88 0.63 0.75 0.85 0.95
6	7.57	10.1	0.98 0.77 0.57 0.57 0.63 0.92 0.63 0.91 0.62 0.96 0.67 0.60 0.63
7	8.07	10.74	0.81 0.74 0.68 0.92 0.79 0.77 0.96 0.64 0.88 0.88 0.69 0.78 0.54
8	7.45	9.93	0.53 0.77 0.89 0.97 0.56 0.78 0.73 0.51 0.67 0.58 0.90 0.66 0.76
9	7.72	10.29	0.58 0.80 0.63 0.83 0.84 0.87 0.73 0.54 0.61 0.96 0.58 0.91 0.77
10	7.84	10.46	1.00 0.54 0.72 0.55 0.98 0.50 0.89 0.91 0.93 0.54 0.70 0.63 0.90

to constraints. By assigning tasks based on technician efficiency, the model identified solutions that minimized downtime. The Branch and bound algorithm used in the MILP model helped manage the numerous possible task allocation scenarios – specifically, 8192 scenarios for this example. This algorithm enabled the model to find the optimal solution efficiently. It is particularly useful for large optimization challenges, where the number of scenarios increases rapidly, i.e. for conditions where availability of skilful manpower is restricted but crucial like small fleet settings where workforce is frequently limited. The analysis of a 50-hour check for the Cessna 172 aircraft illustrates how optimization can reduce task completion times – it achieved approximately a 25% reduction in maintenance time compared to the average completion time in the simulation, based on the optimal time of approximately 7.8 hours compared to the average time. By distributing tasks among technicians with different skill levels, the model balanced the workload effectively, ensuring efficient use of resources. By varying technician efficiency, the model's accuracy remains unaffected. Since the MILP always selects the minimum completion time, such variations influence both manual allocations (and their mean values) and optimized results in a similar way affecting only the numerical outcome, not the validity of the model. It provides a good and easily understandable illustration of the MILP advancement for the small fleet settings.

However, several challenges remain in implementing optimization models in real-world settings. One major issue is the lack of documented task durations and manpower (technicians) efficiencies, which necessitates developing estimation methods based on historical data and averaging previous checks. Although assumptions and the use of randomly generated task durations does not affect the model's ability to determine the optimal solution, the use of generated task durations may not capture all the complexities of actual maintenance scenarios. In addition, applying the model in small aviation organizations may require integration with existing planning systems, which are often manual or fragmented, and may involve basic IT upgrades. Implementation also requires basic familiarity with MATLAB, sufficient to input task data, run the optimization routine, and interpret the resulting task allocation and completion time outputs. Personnel may need introductory training to use such tools effectively. Moreover, Safety Management Systems (SMS), in collaboration with Engineering and Quality departments, can support the evaluation of manpower efficiency and the development of task duration estimates based on historical performance records. These practical considerations represent important directions for applied research and real-world adoption.

Future research could use surveys and historical data to improve task duration estimates and develop better paths for assessing technician efficiency. Additionally, incorporating technician efficiency into the model introduces ethical challenges. In practice, high-performing staff may be consistently assigned more work, potentially leading to overburdening or dissatisfaction. On the other hand, lower-performing individuals may deliberately underperform to avoid workload, creating unfair task distributions. To address this, performance metrics should be supported by a transparent and fair incentive or reward system, ensuring that motivation is preserved and task allocations remain equitable.

References

Cassel, K. W. (2021). Matrix, numerical, and optimization methods in science and engineering. In *Matrix, numerical, and optimization methods in science and engineering*. Cambridge University Press. https://doi.org/10.1017/9781108782333

Comprehensive Airworthiness Organization. (2022). *CESSNA 172S maintenance programme* (LT.CAO.0008). Issue date 2022-07-04.

Derbez, P., & Lambin, B. (2022). Fast MILP models for division property. *IACR Transaction on Symmetric Cryptology*, *2022*(2), 753–753. https://doi.org/10.46586/tosc.v2022.i2.289-321

Dinis, D., & Barbosa-Póvoa, A. P. (2015). On the optimization of aircraft maintenance management. In A. Póvoa, J. De Miranda (Eds), *Operations research and Big Data. Studies in Big Data* (Vol. 15, pp. 49–57). Springer. https://doi.org/10.1007/978-3-319-24154-8_7

Duran, A. S., Gürel, S., & Aktürk, M. S. (2014). Robust airline scheduling with controllable cruise times and chance constraints. *IIE Transactions*, *47*(1), 64–83. https://doi.org/10.1080/0740817X.2014.916457

European Union Aviation Safety Agency. (n.d.-a). *Continuing airworthiness*. Retrieved June 18, 2024, from https://www.easa.europa.eu/en/the-agency/faqs/continuing-airworthiness#category-amp-aircraft-maintenance-programme

European Union Aviation Safety Agency. (n.d.-b). *Safety management system and management system – the integrated approach*. EASA. Retrieved April 29, 2025, from https://www.easa.europa.eu/en/domains/safety-management/safety-management-system-sms

Guo, R., & Wang, Y. (2023). Aircraft assignment method for optimal utilization of maintenance intervals. *Xitong Fangzhen Xuebao / Journal of System Simulation*, *35*(9), Article 13. https://doi.org/10.16182/j.issn1004731x.joss.22-0546

Hasancebi, S., Tuzkaya, G., & Kilic, H. S. (2023). A fuzzy mixed-integer linear programming model for aircraft maintenance workforce optimization. In C. Kahraman, I. U. Sari, B. Oztaysi, S. Cebi, S. Cevik Onar, & A. C. Tolga, *Intelligent and Fuzzy Systems. INFUS 2023. Lecture Notes in Networks and Systems* (Vol. 758, pp. 499–506). Springer. https://doi.org/10.1007/978-3-031-39774-5_56

Johny Ali Firdaus, M., Gharutha, M., & Wangsaputra, R. (2020). Optimization of utilities capacity at aircraft heavy maintenance center using linear programming models. In M. Osman Zahid, R. Abd. Aziz, A. Yusoff, N. Mat Yahya, F. Abdul Aziz, & M. Yazid Abu (Eds), *iMEC-APCOMS 2019. Lecture Notes in Mechanical Engineering* (pp. 115–120). Springer. https://doi.org/10.1007/978-981-15-0950-6_18

Jordan, E., & Azarm, S. (2022). Aircraft maintenance schedule design optimization during a pandemic. In *International Design Engineering Technical Conference & Computers and Information in Engineering* (*3-A*). ASME Digital Collection. https://doi.org/10.1115/DETC2022-90686

Kozanidis, G., Gavranis, A., & Kostarelou, E. (2012). Mixed integer least squares optimization for flight and maintenance planning of mission aircraft. *Naval Research Logistics*, *59*(3), 212–229. https://doi.org/10.1002/nav.21483

Madeira, T., Melício, R., Valério, D., & Santos, L. (2021). Machine learning and natural language processing for prediction of human factors in aviation incident reports. *Aerospace*, *8*(2), Article 47. https://doi.org/10.3390/aerospace8020047

Messac, A. (2015). *Optimization in practice with MATLAB®: For engineering students and professionals*. Cambridge University Press. https://doi.org/10.1017/CBO9781316271391

Muecklich, N., Sikora, I., Paraskevas, A., & Padhra, A. (2023). The role of human factors in aviation ground operation-related accidents/incidents: A human error analysis approach. *Transportation Engineering*, *13*, Article 100184. https://doi.org/10.1016/j.treng.2023.100184

Niu, B., Xue, B., Zhou, T., Zhang, C., & Xiao, Q. (2021). Reorganized bacterial foraging optimization algorithm for aircraft maintenance technician scheduling problem. In Y. Tan & Y. Shi (Eds), *Advances in Swarm Intelligence. ICSI 2021. Lecture Notes in Computer Science* (Vol. 12689). Springer. https://doi.org/10.1007/978-3-030-78743-1_45

Qin, Y., Zhang, J. H., Chan, F. T. S., Chung, S. H., Niu, B., & Qu, T. (2020). A two-stage optimization approach for aircraft hangar maintenance planning and staff assignment problems under MRO outsourcing mode. *Computers and Industrial Engineering*, *146*, Article 106607. https://doi.org/10.1016/j.cie.2020.106607

Ribagin, S., & Lyubenova, V. (2021). Metaheuristic algorithms: Theory and applications. *Studies in Computational Intelligence*, *934*, 385–419. https://doi.org/10.1007/978-3-030-72284-5_18

Sanchez, D. T., Boyacı, B., & Zografos, K. G. (2020). An optimisation framework for airline fleet maintenance scheduling with tail assignment considerations. *Transportation Research Part B: Methodological*, *133*, 142–164. https://doi.org/10.1016/j.trb.2019.12.008

Schulze Spüntrup, F., Ave, G. D., Imsland, L., & Harjunkoski, I. (2021). Integration of maintenance scheduling and planning for large-scale asset fleets. *Optimization and Engineering*, *23*, 1255–1287. https://doi.org/10.1007/s11081-021-09647-7

Sriram, C., & Haghani, A. (2003). An optimization model for aircraft maintenance scheduling and re-assignment. *Transportation Research Part A: Policy and Practice*, *37*(1), 29–48. https://doi.org/10.1016/S0965-8564(02)00004-6

Witteman, M., Deng, Q., & Santos, B. F. (2021). A bin packing approach to solve the aircraft maintenance task allocation problem. *European Journal of Operational Research*, *294*(1), 365–376. https://doi.org/10.1016/j.ejor.2021.01.027