USING MACHINE LEARNING TO OPTIMIZE AIRCRAFT TURN AROUND PROCEDURE

Jordi Sanz Moros[1], Sebastian Andreas Altmeyer [1*]

[1]Universitat Politècnica de Catalunya EETAC - School of Telecom and Aerospace Engineering, 08860 Castelldefels, Spain.

*Correspondence Author :E-mail: sebastian.andreas.altmeyer@upc.edu (https://orcid.org/0000-0001-5964-0203)

Abstract

Efficient aircraft turnaround (TA) is critical for commercial aviation operational productivity. This study investigates the reduction in Total Turnaround Time (TTT) while maintaining strict safety standards by combining deterministic scheduling, partial parallelization, and machine learning. We utilize a deterministic model mapping 13 different operations establishing full-precedence and partial-overlap relationships. Compared to a conservative 84-minute baseline schedule, a baseline optimized model reduces the TTT to 73.5 minutes (a 12.5% improvement) by introducing regulated overlaps among deplaning, cleaning, catering, and boarding. To explore wider operational contexts, a Random Forest Regressor (RFR) was trained on a synthetic dataset of 5000 scenarios generated by varying activity durations and overlap thresholds. The model achieved high predictive accuracy ($R^2 = 0.949$, MAE = 1.85 minutes) and identified a promising unseen configuration (Scenario 928). After applying a vital safety restriction, mandating that boarding only begin after fueling concludes, the final ML-assisted TTT is found at 56.68 minutes. This configuration shaved 17.32 minutes (23.4%) off the baseline optimized schedule while maintaining identical safety parameters. Robustness analysis via 1000 Monte Carlo simulations (±10% duration variance) confirmed schedule stability, yielding a mean TTT of 56.73 minutes and a 95th percentile (P95) of 59.47 minutes. Our findings demonstrate that turnaround performance depends heavily on precedence logic, overlap potential, and network placement rather than just isolated task durations. This study validates deterministic scheduling and machine learning as highly complementary methodologies for optimizing turnaround operations, noting that empirical validation with live operational data remains a necessary step prior to industry deployment.

Keywords*: Aircraft Turnaround, Total Turnaround Time, Deterministic Scheduling, Partial Parallelization, Random Forest Regressor, Monte Carlo Simulation, Aviation Safety Constraints*

NOMENCLATURE

AET	Activity Engagement Time
ATC	Air Traffic Controls
TTT	Total Turnaround Time
ATM	Air Traffic Management
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error
ML	Machine Learning
R^2	Coefficient of determination
TA	Turnaround
RFR	Random Forest Regressor
P95	95th Percentile
IBT	In-block time
OBT	Off-block time
MCS	Monte Carlo Simulations
Ai	Activity i
d_i	Duration of activity i

Copyrights @ Roman Science Publications Ins.

S_i Starting time of activity i
C_i Finishing time of activity i

1. INTRODUCTION

In the highly competitive commercial aviation sector, airline profitability and operational efficiency depend heavily on maximizing aircraft utilization. Because aircraft only generate revenue while airborne, minimizing time spent on the ground is a primary operational objective. Aircraft turnaround, also named as aircraft ground handling are all the necessary procedures to get the aircraft ready for its next flight. The procedure starts during what is referred to as the "on-block time" which is when the aircraft parks after landing, and the wheel chocks are positioned. The off-block time occurs when the aircraft is prepared to exit the parking location and the chocks are taken off. Disembarkation and boarding of passengers, handling of bags and cargo, refueling, cleaning, catering, and ground services are examples of these operations [1]. Various studies [2-5] have been shown that stochastic task durations and precedence relationships are central for the understanding in turnaround reliability. Thereby, main attention has been given to passenger boarding, because it frequently lies on, or at least close to, the critical path for short-haul operations [6,7].

Especially for regional and short-haul operations, an effective aircraft turnaround (TA) [8] is crucial to an airline's overall performance. TA includes a bunch of processes, which run either sequentially or in parallel, based on ability, legal or logistical requirements, and legal constraints. Most studies putting emphasis on the boarding and de-boarding processes [9-11] to identify the critical path as collection of sequential processes consuming the maximum time during the TA [12,13]. In order to handle the growing passenger traffic in the coming years, it is essential that improvements in ground operations be made, particularly in the areas of process dependability and passenger comfort [1]. Any breakdown within this tightly coupled network of activities can propagate across an airline's daily schedule, leading to compounding reactionary delays, financial penalties, and compromised customer service.

Traditionally, ground handling scheduling relies on conservative, sequential baseline frameworks designed to mitigate risk by minimizing overlapping tasks. While these deterministic models ensure adherence to strict safety standards, they often fail to exploit latent operational efficiencies, resulting in prolonged Total Turnaround Time (TTT). Recent advancements in computational intelligence and predictive modeling offer promising alternatives. However, existing literature frequently treats task parallelization and predictive analytics in isolation, creating a research gap in integrated frameworks that can safely compress schedules under highly variable, real-world constraints.

This study addresses this gap by presenting a hybrid optimization framework that merges deterministic network scheduling, partial task parallelization, and Machine Learning (ML). By defining precise full-precedence and partial-overlap logic across 13 distinct ground operations, we model the safety-critical boundaries of the TA ecosystem. In order doing so, we utilize a Random Forest Regressor (RFR) trained on a large synthetic dataset to rapidly navigate complex scheduling configurations and identify high-efficiency operational networks. Finally, we validate the reliability and stability of the optimized schedule against real-world stochastic variability using Monte Carlo Simulations (MCS). The remainder of this paper details the mathematical formulation of our scheduling model, evaluates the predictive performance of the machine learning pipeline, and discusses the operational implications of our optimized TA configurations.

The paper is organized as follows. After the Introduction, section 2 outlines the primary aircraft TA tasks, their durations, dependencies, and potential for parallel execution as well as technical basis on scheduling, critical pathways, bottlenecks, and machine learning. Following, section 3 describes the methodology, including the Random Forest approach, the synthetic dataset, and the deterministic scheduling model. The Baseline and Baseline Optimized schedules are developed in section 4, while the Machine Learning analysis and the final ML-assisted configuration are presented in section 5 including discussion about robustness and limitations along with a comparison of the suggested methods. Following, section 6 provides a quantitative comparison between Baseline and ML model. Finally, section 7 summarizes the key findings and illustrates possible directions for further research.

2. TURNAROUND OPERATIONS

Aircraft TA, also termed aircraft ground handling, encompasses the sequence of operations required to prepare an aircraft for its next flight. This critical process is temporally bounded by two milestones: in-block time (IBT), which begins when the aircraft parks after landing and wheel chocks are positioned, and off-block time (OBT), which occurs when the aircraft is ready to depart and the chocks are removed. Standard operations during this interval include passenger disembarking and boarding, baggage and cargo handling, fueling, cabin cleaning, catering, and essential ground services.

Optimizing this workflow is vital to an airline's overall productivity, especially within regional and short-haul operations characterized by high-frequency schedules. To accommodate projected growth in global passenger traffic, improving ground operations is essential, specifically by enhancing process dependability and passenger comfort.

2.1 Main/Key turnaround activities

Several ground-handling tasks are completed to get the aircraft ready for its next flight after it has been secured at its parking stand. The activities that are later incorporated into the scheduling and ML models serve as a representation of the TA for the purposes of this project. Since taxi-out is considered as a different step, it is not included in this section.

- A1 - Passenger bridge positioning: In order to create a safe connection between the terminal and the passenger cabin, a passenger boarding bridge is placed at the aircraft door at a contact stand. A telescoping tunnel that connects the terminal to the aircraft and facilitates passenger boarding and deplaning is known as a passenger boarding bridge. Passenger stairs, typically in conjunction with buses for transfer between the aircraft and the terminal, provide the same access function at remote stands.
- A2 - Deplaning: This is the procedure by which the passengers of the arriving flight exit the aircraft known as passenger disembarkation, or deplaning. When the aircraft is parked at a remote stand, passengers can leave using passenger stairs or a passenger boarding bridge. The controlled departure of passengers from the airplane and the provision of suitable assistance to passengers with limited mobility are also covered in this activity.
- A3 - Cabin cleaning and services: The passenger cabin is cleaned for the next flight. The interior of the cabin, including the passenger seating area and the elimination of trash from previous occupants, is cleaned and restored to the state that the aircraft operator has requested. Checking the overall state of the cabin and making sure the passenger space is prepared for the next group of passengers are other aspects of cabin services.
- A4 - Catering services: Catering services include loading the supplies needed for the next service and removing the catering equipment and supplies used on the arrival flight. These goods could include food, drinks, galley supplies, and catering trolleys, depending on the airline and trip.
- A5 - Cargo unloading: The removal of both commercial cargo and passenger baggage from the arriving flight is referred to as unloading. After gaining access to the airplane holds, ground staff move the contents to the proper ground-support equipment so they can be transported to the cargo facilities or baggage handling system.
- A6 - Cargo loading: Loading consist of placing the baggage of the incoming passengers into the aircraft holds the necessary loading guidelines must be followed while distributing and securing the products
- A7 - Water service: This procedure consists in restocking the aircraft's potable water tanks with water for drinking and other onboard personal needs is known as potable water servicing. Dedicated equipment connected to the appropriate aircraft interface is used to carry out the service
- A8 - Toilet service: Toilet services include clearing the collected waste from the lavatory system and getting it ready for the next flight.
- A9 - Fueling: The procedure of providing the gasoline needed for the next flight is known as aircraft fueling. A fueling truck or a dispenser linked to an airport hydrant system might offer the service. The complete process involves connecting the fueling machine, moving the necessary amount of gasoline, and disconnecting it after the fuel uplift is finished.
- A10 - Boarding: The procedure by which passengers arrives to the aircraft and settle into their designated seats is

known as boarding. Passengers find their seats within the cabin and store their hand luggage underneath or in the overhead bins. Before the airplane doors are closed, the cabin crew oversees the procedure and completes the required passenger and cabin inspections.

- A11 - Passenger bridge removal: The passenger boarding bridge is detached from the aircraft door and put back in its secure location prior to take-off. The corresponding procedure at remote stands is to clear the aircraft area of the related equipment and remove the passenger steps. Verifying that the aircraft doors have been correctly closed and that the external equipment has been cleared is another step also included in the removal procedure.
- A12 - Pushback: Pushback is the procedure where the aircraft is moved backwards from a nose-in parking position to be ready for next phase. As part of the leaving process, the engine may also be started. Pushback is the last action taken into account in the TA model prior to the start of the independent taxi-out phase.

2.2 Activity durations and operational impact

The duration of an aircraft TA operation should not be conceptualized as a static parameter. Instead, it is dynamically governed by multi variables, including aircraft type, passenger load factor, cargo and baggage volume, gate or stand configuration, airline business model, available ground-handling resources, and specific servicing requirements (A1-A12). Furthermore, the total duration allocated to any given task extends beyond its pure execution window; it encompasses the time required to position, connect, detach, and remove the corresponding ground-support equipment. In literature, this comprehensive operational interval is formally designated as the Activity Engagement Time (AET) [1].

Because TA operations comprise a complex network of concurrent and sequential processes, the TTT is never simply the cumulative sum of individual task durations. Rather, TTT is dictated strictly by the longest path through the scheduling network, the critical path. Consequently, a long-duration task executed fully in parallel may have negligible impact on the overall schedule if it finishes before the aircraft is ready for departure. Conversely, a brief, non-parallelizable task that serves as an unavoidable predecessor to passenger boarding, door closure, or pushback can directly significant delay the off-block milestone.

To illustrate these dynamics, the operational parameters for an Airbus A320 operating under a full-service carrier scenario, adapted from the foundational work by Scardaoni et al. [14,15], are listed in Table 1. The activity durations were mathematically derived using the specific passenger, containerized cargo, fueling, and catering throughput rates reported by the authors. Crucially, these parameters represent a single, isolated operational configuration and should not be interpreted as generalized TA standards.

Table 1 Range durations and operational impact of TA activities adapted from Scardaoni et al. [14]

Acivity	Typical duration (minutes)	Key factors that could cause delays	Operational impact
Passenger bridge positioning	1-3	Ground crew availability or malfunction of the bridge.	Low but it's the activity that starts.
Deplaning	10-20 (narrow body) 20-30 (wide boy)	Large number of passengers, passengers that requiring assistance and if there is no bridge positioning, availability of busses.	Moderate. Some activities may depend.
Cabin cleaning	10-20	Aircraft size (more size, more cleaning is needed) and insufficient number of workers.	Moderate to hight, some activities may depend directly.
Catering services	10-30	Truck availability	Low
Cargo Unloading	10-30	Quantity of baggage and ground crew performance	High, cargo loading can't start unless this activity is partially finished.
Cargo loading	10-30	Quantity of baggage and ground crew	High

		performance	
Fueling	15-30	Required fueling, fuel truck availability and safety restrictions	High
Water Services	5-10	Coordination with other trucks	Low
Toilet services	5-10	Equipment availability	Low
Boarding	15-30	Large number of passengers, passengers that requiring assistance, hand luggage and late passengers	Very High, often Boarding is in critical path
Passenger bridge removal	1-2	Door closure and ground crew confirmation	Low, but it is mandatory predecessor of pushback
Pushback	3-5	ATC clearance	High delays directly affect to taxi-out and departure

Several of the operational ranges established in this study correlate directly with the quantitative metrics defined by Scardaoni et al. [14]. For instance, the abbreviated intervals allocated for passenger boarding bridge operations are corroborated by the reference value of approximately 2.0 minutes reported for equipment positioning and removal. Similarly, the reference Airbus A320 scenario assumes a payload of seven cargo containers; at a baseline handling rate of 1.5 minutes per container, this yields a foundational duration of roughly 10.5 minutes before accounting for additional ground-handling variables.

For the fueling phase, a volumetric flow rate of 1.25 m^3/min applied to a reference volume of 20 m^3 translates to a fuel transfer window of approximately 16.0 minutes. This mathematically aligns with the 15–25-minute fueling envelope utilized in this study. Conversely, broader duration intervals were designated for passenger boarding and deboarding. These operations are modeled using variable passenger throughput rates, which fluctuate dynamically based on the total passenger complement and the number of active cabin doors.

While these operational references anchor the models within realistic orders of magnitude, the synthetic dataset is intentionally not calibrated to represent an Airbus A320 exclusively, nor is it restricted to a specific airport, airline, or ground-handling configuration. Instead, the defined parameters delineate a generalized turnaround design space. By expanding several operational intervals beyond the boundaries of the baseline A320 reference case, the synthetic analysis can systematically explore a significantly broader spectrum of operational scenarios and network variations.

2.3 Dependencies and parallelism/overlaps

TA dependencies are categorized primarily by their operational and physical origins. Physical dependencies dictate that an action requires direct aircraft access or the completion of a prior physical procedure. For example, passenger deplaning cannot commence until the boarding bridge is positioned, and the bridge cannot be removed until boarding concludes, doors close, and the surrounding area is cleared. Similarly, cabin cleaning requires deplaning to be sufficiently advanced, while subsequent boarding depends on adequate cleaning progress. These sequential workflows do not always require absolute task completion; under specific configuration rules, a succeeding task can initiate once a defined progress threshold is reached. Table 2 list the different activities with indication of potential parallelization.

Table 2 Main dependencies, type of dependency and parallelism for each activity

Acivity	**Main dependency**	**Possible to parallelize?**	**Dependency type**
Passenger bridge positioning	Aircraft parked correctly	No	Physical
Deplaning	Passenger bridge positioned	Yes	Physical and operational
Cabin cleaning	Deplaning sufficient advanced	Conditional	Physical and operational
Catering services	Deplaning sufficient advanced	Conditional	Operational
Cargo Unloading	Aircraft parked correctly and	Yes	Operational

	handling equipment positioned		
Cargo loading	Cargo loading sufficiently advanced	Conditional	Operational
Fueling	Aircraft parked correctly and necessary equipment available	Yes	Safety and operational
Water Services	Service equipment available	Yes	Operational
Toilet services	Service equipment available	Yes	Operational
Boarding	Cabin sufficiently prepared	Yes	Physical and operational
Passenger bridge removal	Boarding completed and doors closed	Yes	Physical
Pushback	Passenger bridge removal completed, and ground crew cleared the area	No	Safety and operational

Alternatively, operational coordination and safety dependencies govern concurrent workflows. While fueling, catering, cargo handling, and water/waste servicing can execute in parallel, their simultaneous execution is strictly limited by ramp space, ground-support vehicle positioning, and safety protocols. Finally, aircraft movement is completely restricted until all critical ground-handling tasks conclude and all personnel and equipment safely clear the aircraft footprint.

2.4 Bottlenecks and critical activities

While closely interrelated in aircraft TAs, bottlenecks and critical activities are conceptually distinct. A critical activity lies on the critical path, meaning any delay directly postpones the aircraft's departure time. Conversely, a bottleneck is an operational constraint or task that restricts the throughput or initiation of subsequent activities. Consequently, the longest-duration task is not inherently the most critical; its operational impact depends entirely on its capacity for parallel execution alongside concurrent workflows.

Furthermore, bottlenecks frequently arise from operational variables rather than nominal task durations. Factors such as delayed ground-support equipment arrival, staffing shortages, restricted ramp space around the aircraft, service vehicle interference, or safety constraints on concurrent operations can severely diminish parallel execution. Under these stochastic conditions, a non-critical task can quickly shift onto the critical path, dictates the overall success of the TA. Therefore, the exact composition of critical tasks is dynamically governed by each schedule's unique durations, dependencies, and operational limits, the specific bottlenecks and critical paths derived from the proposed models are analyzed in subsequent chapters.

2.5 Taxi-out as extra phase

Following the completion of pushback, the aircraft transitions from the parking stand toward the runway during the taxi-out phase. The duration of this phase is highly variable and depend on the particular airport. It is primarily dictated by runway distance, taxiway availability, airport traffic congestion, and Air Traffic Control (ATC) directives.

Although aircraft TA conventionally concludes at the off-block milestone, this study incorporates taxi-out as a distinct, sequential phase/activity (A13) following the core ground-handling procedures. Because taxi-out duration is predominantly governed by external airport traffic and active surface conditions rather than gate operations, it is modeled independently from the standard internal ground-handling network/activities (A1-A12).

2.6 Scheduling problems in airport operations

The challenge of determining the optimal timing for a specific set of operations, subject to predefined execution constraints, constitutes the scheduling problem. Within this framework, tasks represent individual aircraft TA ground-handling operations, each bounded by a specific start and finish time. The operational dependencies between these tasks are modeled as precedence constraints, which dictate whether activities must execute sequentially or can run concurrently.

The primary objective is to maximize parallel processing to reduce the overall TTT without violating strict operational or safety mandates. However, the capacity for simultaneous activity surrounding the aircraft is heavily bounded by physical and logistical limitations. Specifically, the availability of ground-support equipment, labor allocation, apron space constraints, and spatial interference between service vehicles all restrict the achievable level of parallelization and directly impact the overall turnaround timeline.

2.7 Critical path and bottleneck analysis

The sequence of dependent tasks that dictates a schedule's overall completion time is defined as the critical path. Activities along this path possess zero scheduling flexibility; any postponement directly extends the TTT. Identifying the critical path is therefore essential for determining which operations exert the greatest leverage over the departure time and where operational modifications will yield the most impacting time savings [1,16]. Conversely, a bottleneck represents a specific activity or operational constraint that obstructs the advancement of subsequent tasks. While a bottleneck can be a function of task duration, it is frequently induced by resource limitations, such as personnel shortages, inadequate ground-support equipment, restricted apron space, or ramp vehicle interference [17].

Although closely interrelated, these two concepts describe distinct dimensions of TA performance. Critical path analysis isolates the specific sequence of tasks that mathematically bounds the total duration, whereas bottleneck analysis identifies the underlying resource or physical constraints that disrupt the fluid flow of operations. Consequently, integrating both perspectives provides a comprehensive framework for strategic scheduling adjustments and resource allocation to compress TTT.

3. METHODOLOGY

The research methodology employs a hybrid approach that couples a rule-based deterministic scheduling framework with data-driven predictive modeling to analyze the aircraft TA performance [18,19]. This integrated model captures the core ground-handling operations (A1-A12) defined in previous chapters and incorporates taxi-out (A13) as a distinct final phase. The workflow is executed in three sequential stages:

- Stage 1: Deterministic Scheduling Formulation

The initial phase establishes a rule-based deterministic scheduling model. Given a specific vector of activity durations, partial-overlap thresholds, and precedence relationships, the model mathematically solves for the exact start and finish times of each task. From this network configuration, it extracts the TTT, identifies the critical path, and isolates operational bottlenecks [19,20]. The framework is classified as deterministic because any identical combination of input durations and precedence constraints invariably yields a consistent, reproducible schedule.

- Stage 2: Synthetic Dataset Generation

To explore a comprehensive design space, the deterministic scheduling logic is applied to generate a large-scale synthetic dataset of unique TA scenarios. While the underlying physical and safety constraints of the network remain invariant, the 13 individual activity (A1-A13) durations and 4 partial-dependency thresholds are systematically varied within predefined operational ranges. Rather than relying on empirical observations from live airport operations, this dataset maps the multi-dimensional behavior and sensitivity of the scheduling model across diverse parameter configurations.

- Stage 3: Predictive Modeling and Optimization

In the final phase, the synthetic dataset is utilized to train and evaluate a RFR. Rather than generating a schedule directly, the ML model maps the non-linear relationships connecting the 17 input variables (13 activity durations and 4 overlap thresholds) to the resultant TTT calculated by the deterministic model. Once trained and validated, this regressor functions as a rapid predictive tool to scan unseen test cases, identifying exceptionally high-efficiency configurations (such as Scenario 928) for targeted downstream optimization within the deterministic framework.

3.1 Definition of activities and variables

As noted before, the AT process is represented by thirteen activities. The first twelve phases are dedicated to ground handling and departure preparation procedures, with taxi-out included as a final step. Table 3 lists the IDs with the

corresponding activity. To simplify the expressions, each activity is labeled as Ai, where i is a number between 1-13.

Table 3 Project IDs and corresponding activities

Project ID	Activity
A1	Passenger bridge positioning
A2	Deplaning
A3	Cabin cleaning
A4	Catering
A5	Cargo unloading
A6	Cargo loading
A7	Water service
A8	Toilet service
A9	Fueling
A10	Boarding
A11	Passenger bridge removal
A12	Pushback
A13	Taxi-out

For each activity we will consider three basic variables:

- Duration of each activity in minutes (d_i)
- Start/Initiation time of an activity (S_i)
- End/Finishing time of an activity (C_i)

Therefore, the total completion time is:

$$C_i = S_i + d_i. \quad (1)$$

Consequently, the TTT, is the Ending time of the last activity minus the starting time of the first activity (normally this value will be 0 as the starting time is t = 0).

$$TTT = C_{13} - S_1. \quad (2)$$

3.2 Deterministic Scheduling Model Formulation

The core computational framework of this study relies on a rule-based deterministic scheduling model to map and evaluate TA timelines. Given a defined vector of activity IDs, durations, and partial-overlap thresholds, the model applies a fixed network logic to determine the exact start and finish times for each operation. From the resulting schedule, the framework algorithmically extracts the critical path, operational bottlenecks, and the TTT. The model is classified as deterministic because a given set of input values and precedence constraints will invariably yield an identical, reproducible scheduling network.

This uniform deterministic architecture is applied sequentially across three evolutionary stages of the research to identify increasingly high-efficiency TA configurations:

- The Baseline Model: Represents the initial implementation, utilizing standard nominal activity durations under a conservative, fully sequential scheduling configuration.
- The Baseline Optimized Model: Utilizes the identical activity durations as the baseline but introduces calibrated partial-overlap thresholds to permit parallel execution between key concurrent workflows (e.g., deplaning, cleaning, catering, and boarding).
- The ML-Assisted Model: Maintains the same underlying network logic, but populates the model with a highly favorable activity-duration profile (here Scenario 928) discovered by the RFR. The deterministic scheduling model is then deployed downstream to optimize the overlap thresholds for this specific profile.

The progressive objective of this multi-stage workflow is to systematically uncover superior TA configurations. By first maximizing parallel execution within a static baseline framework, and subsequently leveraging machine learning to navigate a broader operational design space, the framework successfully compresses TTT while strictly adhering to the fundamental scheduling and safety restrictions embedded within the deterministic network.

3.3 Full precedence constraints

The constraints are restrictions that state when one activity is dependent on another, the subsequent activity cannot begin until the prior one is completed. Relationships where the predecessor activity must be finished completely before the successor activity can start are known as full-precedence constraints. Every time the deterministic scheduling model computes a TA schedule, these constraints are implemented as part of the fixed dependence structure.

Table 4 Predecessors and successors of activities

Predecessor	Successor	Reason
A1	A2, A5, A7, A8, A9	Schedule can only start if passenger bridge has been correctly positioned
A10	A11	Boarding must be completed before passenger bridge removal starts
A6, A7, A8, A9, A11	A12	Every activity needs to be finished before the aircraft starts moving
A12	A13	Taxi-out can only start if pushback is completed

All associated precedence restrictions must be finished when an activity has several predecessors as illustrated in Table 4. As a result, the activity cannot begin until the final necessary predecessor has been finished unless a partial overlap threshold is used. In that case the percentage of finished activity is determined by this threshold. The deterministic scheduling model's strictly sequential relationships are defined by this set of requirements.

3.4 Partial overlap constraints

In real-world ground handling, network dependencies do not inherently demand the absolute completion of a predecessor task before a successor can initiate. In many operational contexts, once a preceding action progresses to a sufficient degree, the subsequent activity can safely begin. The scheduling framework incorporates partial overlap constraints to mathematically model these concurrent interactions (Table 5).

Table 5 Activities with partial overlap

Identifier	Predecessor	Successor	Reason
A2 →A3	A2	A3	Cleaning services are able to start once deplaning is sufficiently advances
A2 →A4	A2	A4	Catering also can start before deplaning is fully completed
A3 → A10	A3	A10	Boarding may begin once the cabin cleaning has reached a sufficient level
A5 →A6	A5	A6	At the same time, you are unloading the cargo you can start loading again with new baggage

A partial overlap constraint is governed by a defined completion threshold, which denotes the minimum cumulative proportion of a predecessor task that must be completed before the successor activity can be triggered. Structurally, a threshold value of 100% represents a conventional full-precedence relationship (full-precedence constraints). Conversely, lowering this threshold value introduces a progressive degree of parallel execution into the network. By applying these mathematical criteria, the model dynamically captures a diverse spectrum of operational parallelization without altering individual task durations. Instead, by systematically allowing compatible, non-interfering tasks to execute concurrently, the overall network path is compressed, thereby minimizing TTT. Within the computational

scheduling model, these threshold values function as highly sensitive, adjustable input parameters.

3.5 Synthetic dataset generation

To evaluate the proposed methodology in the absence of empirical operational data, a large-scale synthetic dataset was constructed. The objective was to generate a robust distribution of scenarios featuring varied activity durations and overlap conditions while strictly preserving the underlying deterministic scheduling logic. This simulated database provides the foundational training data required by the ML pipeline and enables systematic sensitivity testing of the deterministic network across a diverse spectrum of operational conditions.

The resulting synthetic dataset contains 5000 distinct TA scenarios across 22 operational variables. Each row represents a discrete, fully resolved TA configuration. For every individual scenario, the 13 activity (A1-A13) durations and 4 partial-overlap thresholds (Table 5) are randomly sampled from their predetermined engineering ranges. The deterministic scheduling framework then ingests these stochastic parameters as inputs to algorithmically resolve the network timeline and compute the corresponding operational outputs (such as critical paths and final timelines). This controlled synthetic architecture serves as a rigorous baseline to validate the performance and boundaries of the integrated scheduling model. The operational 22 variables can be divided in four groups as listed in Table 6.

Table 6 Group of variables of the synthetic dataset

Variable group	Number of variables	Description
Identifier	1	An identifier assigned to each scenario
Durations	13	The durations of the activities A1 to A13
Overlap thresholds	4	Minimum completion level required before a dependent activity can start
Outputs	4	TTT, bottleneck task, bottleneck name and critical path

In addition, the four overlap thresholds will have ranges (%) as listed in Table 7.

Table 7 Range of Thresholds

Identifier	Predecessor	Successor	Range (%)
A2 →A3	A2 - Deplaning	A3 - Cabin cleaning	60 - 100
A2 → A4	A2 - Deplaning	A4 - Catering	60 - 100
A5 → A6	A5 - Cargo unloading	A6 - Cargo loading	50 - 100
A3 → A10	A3 - Cabin cleaning	A10 - Boarding	70 - 100

The lower bounds of the overlap threshold ranges are strategically designated to define a controlled spectrum of partial parallelization while preventing physically unconstrained concurrence between dependent tasks. Specifically, a lower threshold of 60% is established for both the deplaning-to-cleaning (A2→A3) and deplaning-to-catering (A2→A4) interactions. Meanwhile the cleaning-to-boarding (A3→A10) the lower threshold is considered at 70%. In contrast, the cargo unloading-to-loading sequence (A5→A6) is permitted to explore a broader degree of concurrent execution, extending down to a lower bound of 50%. Conversely, an upper threshold bound of 100% represents a strict full-precedence constraint, dictating that a successor activity cannot initiate until its predecessor has concluded.

For every generated scenario in the synthetic dataset, the computational model executes the full-precedence and partial-overlap logic outlined in the former sections. Based on the stochastically sampled activity durations and overlap criteria, the model solves the network timeline, isolates the resulting bottleneck and critical path, and computes the absolute TTT. The input variables and their corresponding scheduling outputs are co-indexed and stored together within the database.

Crucially, because this dataset is entirely synthetic and independent of direct empirical observations from an active

airport, it reflects the mathematical behavior and boundary dynamics of the proposed scheduling logic under diverse configurations. Consequently, interpreting the dataset - and evaluating the predictive performance of the ML model trained on it - requires strict adherence to these foundational scheduling rules and operational assumptions. Finally, it should be noted that the extra safety mandate requiring passenger boarding (A10) to begin only after fueling (A9) concludes is intentionally omitted during the initial synthetic data generation phase; this restriction is introduced downstream during the final operational review phase.

3.6 Machine Learning model

The synthetic dataset serves as the foundation for training a supervised ML regression model. Because the target variable, TTT, is a continuous numerical metric, the problem is explicitly formulated as a regression task. The primary objective of this predictive pipeline is to mathematically capture the non-linear correlations and complex interactions mapping the operational input factors to the final network duration resolved by the deterministic model.

Crucially, the ML approach does not replace the rule-based deterministic scheduling model. Instead, it functions as a highly efficient meta-model that provides a rapid predictive approximation of the scheduling framework's behavior. By mapping the multidimensional design space, the regressor efficiently sifts through thousands of variable permutations to isolate highly favorable operational configurations with the latent potential to achieve minimal TA times. Once a promising candidate configuration is identified by the ML pipeline, it is re-ingested into the deterministic scheduling model for exact network resolution, structural constraint validation, and final downstream threshold optimization.

3.6.1 Input and target variables

The input feature space for the machine learning model consists of 17 independent operational variables: the 13 ground-handling durations and the four partial-overlap thresholds defined in the preceding sections. For each scenario in the synthetic dataset, these 17 parameters are ingested by the deterministic scheduling model, which applies the network priority logic and partial-overlap requirements to establish the definitive start and finish times for each operation. The resulting TTT is then stored within the database to serve as the continuous target variable for the supervised regression task.

To ensure model generalizability and prevent data leakage, strict feature selection is applied. Metadata variables, such as the unique scenario "identifier," are excluded from the training process as they contain no predictive value. Furthermore, operational outputs generated by the deterministic network—including the bottleneck name, bottleneck task ID, and specific critical path composition—are intentionally omitted. Because these metrics are downstream causal derivatives of the 13 durations and four thresholds, including them in the feature space would introduce severe multi co-linearity and redundancy.

During the model evaluation phase, the predictive performance of the Random Forest Regressor is rigorously assessed. The predicted TTT values generated by the machine learning model are validated against the true reference TTT values calculated by the rule-based deterministic scheduling model. This comparison quantitatively measures the regressor's fidelity in learning, generalizing, and replicating the non-linear behavioral dynamics of the underlying scheduling network framework.

3.6.2 Random Forest Architecture and Configuration Selection

To capture the multi-dimensional complexities of the TA network, a RFR was selected as the core ML model. The Random Forest is an ensemble learning method constructed from a large collection of independent decision trees. Each individual tree generates a distinct continuous prediction, and these outputs are subsequently aggregated, typically via arithmetic averaging, to establish the final regression target.

This ensemble approach is ideally suited for this problem due to the inherently non-linear relationships that govern the scheduling design space. The absolute TTT is not a simple linear sum of inputs; instead, it is a dynamic product of task durations, structural dependencies, parallel execution overlaps, and sequence interactions. For instance, a temporal fluctuation in a task executing concurrently alongside other workflows may have zero impact on the final timeline, yet that exact same fluctuation will cause a 1-to-1 shift in TTT if the task falls onto the critical path. The RFR effectively models these step-functions and non-linear network interactions natively, without requiring explicit

mathematical formulation within the predictive pipeline.

To rigorously train and evaluate the regressor, the 5000 generated synthetic scenarios are split into distinct training and testing subsets. The training partition is utilized to fit the ensemble parameters and map the underlying connections between the 17 operational variables and the TTT. Conversely, the test partition consists of entirely unseen holdout configurations used exclusively to assess model generalization and verify how accurately the regressor replicates the deterministic network's true behavior.

Following this validation step, the trained Random Forest is deployed to predict the TTT across all remaining test permutations. The specific configuration that yields the lowest predicted TTT is selected as a promising high-efficiency candidate for downstream optimization (in our case Scenario 928). While freezing the 13 stochastically discovered activity durations of this candidate scenario, the deterministic scheduling model is reintroduced to systematically sweep and optimize the 4 partial-overlap thresholds. This hybrid coupling leverages the predictive power of the Random Forest to rapidly discover optimal operational conditions, while relying on the rule-based model to ensure absolute schedule feasibility.

3.7 Evaluation Framework

The evaluation framework reflects the distinct, yet complementary roles of the rule-based deterministic scheduling model and the RFR within the proposed framework. While the deterministic scheduling model is deployed to calculate exact network timelines and evaluate structural schedule feasibility, the ML pipeline is appraised strictly on its capacity to accurately predict the continuous TTT of entirely unseen synthetic scenarios.

Consequently, the evaluation process is bifurcated into two distinct phases:

1. Statistical Model Validation: Standard statistical regression metrics are utilized to quantify the predictive accuracy and generalization performance of the RFR on the holdout test subset.
2. Operational Performance Comparison: A downstream comparative analysis is conducted to contrast the final TA configurations yielded by the hybrid ML-assisted methodology against the traditional Baseline Optimized schedules under identical safety parameters.

3.7.1 ML Performance Metrics

The predictive performance of the RFR on the test subset is validated using three distinct statistical indicators: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2).

Utilizing this triad of metrics provides a comprehensive assessment of model fidelity. While R^2 assesses how effectively the model captures the underlying variance of the TTT across heterogeneous operational scenarios, MAE and RMSE quantify the magnitude and distribution of the residual prediction errors.

Mean Absolute Error (MAE)

The MAE calculates the average magnitude of the absolute residuals between the true reference TTT derived from the deterministic model and the predicted TTT generated by the RFR. The metric is mathematically formulated as:

$$\mathrm{MAE} = \frac{1}{n}\sum|y_i - \hat{y}_i|, \quad (3)$$

where y_i represents the deterministic reference value, $\hat{y}_i$ denotes the predicted value, and n is the total number of test scenarios.

A primary advantage of MAE is its direct, linear interpretability. Because TTT is measured in minutes, the resulting MAE is expressed in identical temporal units, providing an intuitive baseline indicator of the model's practical, day-to-day prediction error. Lower MAE values indicate that the regressor's estimates are consistently close to the true network-resolved durations.

Root Mean Squared Error (RMSE)

Like MAE, the RMSE measures the discrepancy between the reference and predicted TTT values in minutes. However, because the residuals are squared before their average is computed, the RMSE penalizes larger deviations more heavily. The metric is defined as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum(y_i - \hat{y}_i)^2}. \quad (4)$$

This quadratic weighting is operationally significant in aviation scheduling. While a model may maintain an acceptable average error profile (reflected in a low MAE), it could occasionally generate severe under-estimations or over-estimations for specific configurations. In commercial ground-handling operations, large individual errors create substantial logistical risks; thus, minimizing the RMSE ensures the model's predictive reliability across all edge-case scenarios.

Coefficient of Determination (R^2)

The R^2 metric quantifies the proportion of total variance in the true TTT target values that can be successfully explained by the operational input features via the Random Forest architecture. It is expressed as:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}, \quad (5)$$

where $\bar{y}$ is the mean of the true reference values.

An R^2 value approaching 1 signifies that the model accurately captures how the overall timeline scales across diverse operational contexts. This metric is critical because the goal of the regressor extends beyond predicting a static baseline mean; it must dynamically map how TTT fluctuates in response to complex, non-linear interactions between the 13 activity durations and 4 overlap thresholds. Therefore, a high R^2 serves as statistical proof that the Random Forest has successfully mapped the underlying logic of the scheduling network.

3.7.2 Comparison between ML and Baseline models

The final phase of the framework focuses on a rigorous operational evaluation of the Baseline Optimized and ML-assisted TA configurations. Although these two configurations are derived via distinct methodologies, both are ultimately resolved using the identical underlying deterministic scheduling model. The Baseline Optimized configuration preserves the nominal baseline activity durations while introducing calibrated partial overlaps to compress the timeline. Conversely, the ML-assisted configuration populates the scheduling framework with a highly favorable activity-duration profile identified by the RFR, and subsequently deploys the deterministic model to dynamically refine the corresponding overlap thresholds.

To guarantee a scientifically consistent and unbiased comparison, a critical aviation safety constraint is uniformly imposed on both final configurations: the initiation of passenger boarding (A10) is strictly prohibited until fueling (A9) has concluded. This standardization yields two safety-adjusted schedules operating under identical constraint boundaries. By eliminating confounding operational variables, this controlled comparative framework isolates and quantifies the exact impact in using ML to navigate a broader spectrum of operational conditions. Table 8 lists all variables that we will evaluated in the following sections.

Table 8 Variables for comparison between ML and deterministic model

Variable / Criteria	Explanation
TTT	Both approaches will lead to an ideal TTT
Prediction accuracy	Depending on the values of the ML metrics any comparison between models needs be considered
Constraint compliance	Whether operational, precedence and partial overlaps are respected
Robustness	Models behavior when activity durations or overlap conditions change
Safety and operational feasibility	Whether it remains compatible operational and safety requirements

4 BASELINE MODEL

4.1 Baseline assumptions

The baseline model provides as a cautious baseline for assessing the impact of increasing parallelism in the following optimized scheduling model. The same 13 activities listed in previous sections are taken into consideration, with activity A1–A12 standing in for the ground-handling procedure and A13, taxi-out, added as a post pushback phase.

Based on the characteristics provided by Scardaoni et al. [14], the baseline activity durations were chosen as representative values within the operational ranges covered in previous sections. Activity durations are fixed and

deterministic for this baseline scenario. Table 9 lists the exact chosen time durations for the corresponding activities.

Table 9 Baseline Model parameters

Project ID	Activity	Duration
A1	Passenger bridge positioning	2
A2	Deplaning	15
A3	Cabin cleaning	15
A4	Catering	20
A5	Cargo unloading	20
A6	Cargo loading	20
A7	Water service	8
A8	Toilet service	8
A9	Fueling	20
A10	Boarding	30
A11	Passenger bridge removal	2
A12	Pushback	5
A13	Taxi-out	15

4.2 Baseline Gantt chart

Figure 1 shows the Gantt chart of the Baseline Model illustrating the initial schedule generated by the deterministic model. The 13 operational activities are mapped chronologically alongside their respective start and finish milestones, delineating the hybrid sequence of concurrent and linear workflows that characterize the standard TA process.

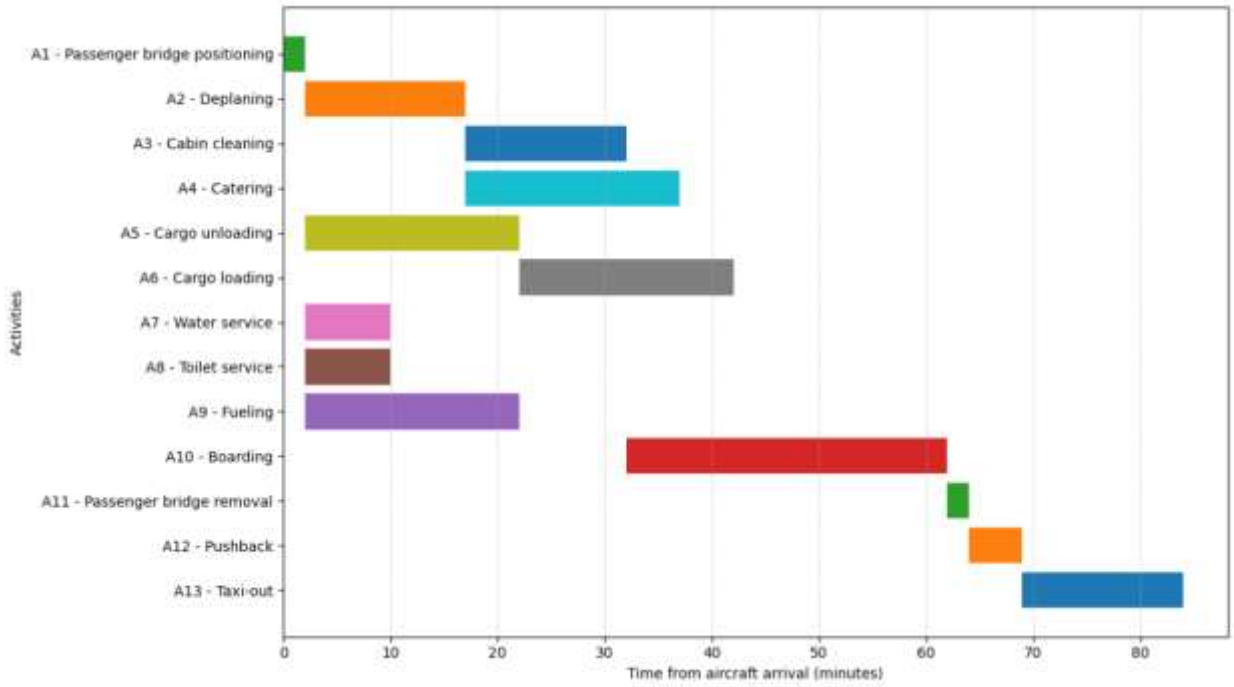


Figure 1 Gantt chart of the Baseline model

The network initiates with passenger bridge positioning (A1), which serves as the immediate predecessor for

passenger deplaning (A2). Once deplaning commences, cargo unloading (A5), potable water servicing (A7), lavatory servicing (A8), and fueling (A9) are permitted to execute concurrently. Within this baseline configuration, the dependencies designated for future partial-overlap testing are restricted to full-precedence constraints, requiring absolute completion of all predecessor tasks. Consequently, cabin cleaning (A3) and catering (A4) are prohibited from initiating until passenger deplaning (A2) concludes, and cargo loading (A6) can only begin once cargo unloading (A5) is fully realized. Similarly, passenger boarding (A10) is constrained to start only after cabin cleaning (A3) has entirely finished. This configuration corresponds to a strict 100% completion threshold across all dependent junctions, establishing a conservative operational reference baseline.

The scheduling network also demonstrates that several parallel ground-handling operations run independently of the critical passenger sequence. Cargo loading, water servicing, lavatory servicing, and fueling proceed concurrently without directly restricting the initiation of boarding. However, to guarantee ramp safety and ensure the aircraft does not begin moving while ground-servicing procedures are active, all of these workflows maintain a strict full-precedence relationship with the pushback phase; they must safely conclude before the aircraft footprint is cleared.

Once passenger boarding is completed, the final ground-handling phase terminates with the removal of the passenger bridge (A11), immediately triggering the pushback sequence (A12). Finally, the taxi-out phase (A13) is introduced into the computational model as the terminal, sequential operation leading to active runway departure.

4.3 Baseline TTT

The TTT of the baseline schedule $TTT_{baseline}$ is calculated from the initiation of passenger boarding bridge positioning (A1) to the conclusion of the terminal taxi-out phase (A13), yielding a cumulative duration of 84 minutes.

Within this framework, the core ground-handling operations (A1 through A12) require 69 minutes to execute. The remaining 15 minutes are consumed entirely by the taxi-out phase (A13) to achieve the overall 84-minute baseline profile. Maintaining this explicit distinction throughout downstream optimization phases is critical to avoid conflating external airfield variables with the intrinsic performance of the gate-side ground-handling network.

The duration of the taxi-out phase is predominantly governed by stochastic airfield variables rather than the operational efficiency of the preceding gate activities. Surface traffic congestion, taxiway geometry, ATC sequencing, and runway departure queues can all introduce significant variance to the time elapsed between pushback and takeoff. Prior literature has explicitly underscored the compounding influence of airport surface congestion on aircraft taxi times, confirming the mathematical necessity of modeling this terminal phase independently from the localized ground-handling network at the stand [21].

4.4 Baseline Bottlenecks and Critical Path Analysis

The structural analysis of the Baseline schedule indicates that the primary operational constraint is dictated entirely by passenger-related workflows. The definitive critical path is algorithmically resolved as the sequence:

$$A1 \rightarrow A2 \rightarrow A3 \rightarrow A10 \rightarrow A11 \rightarrow A12 \rightarrow A13 \quad (6)$$

This critical sequence encompasses bridge positioning (A1), passenger deplaning (A2), cabin cleaning (A3), passenger boarding (A10), passenger bridge removal (A11), pushback (A12), and eventually the terminal taxi-out phase (A13). Because these core activities are linked in a strict, fully sequential linear chain within the baseline network, any operational delay incurred by a single task propagates downstream with a 1-to-1 negative impact on the final aircraft departure time.

Conversely, the primary operational bottleneck within this scheduling network is isolated as the passenger boarding phase (A10). With a nominal duration of 30 minutes, it constitutes the longest-duration task situated along the critical path, representing the narrowest throughput constraint in the baseline TA life-cycle.

4.5 Principles of Deterministic Schedule Optimization

The optimization of the deterministic scheduling model is structurally predicated on maximizing the level of parallel execution across the TA network. In contrast to conventional management methodologies that attempt to accelerate or compress the absolute duration of individual ground-handling tasks, the proposed framework preserves nominal activity durations. Instead, it systematically minimizes non-value-added buffer intervals and idle waiting periods between dependent workflows.

While the baseline configuration operates under conservative full-precedence assumptions, i.e. mandating the absolute completion of a predecessor activity before a successor can initiate, the optimized framework relaxes these linear constraints by introducing the calibrated completion thresholds as discussed before. By establishing completion thresholds below 100%, the model permits a successor activity to trigger as soon as its predecessor has progressed to a mathematically defined operational checkpoint. This relaxation compresses the cumulative network path without compromising the baseline safety limits or structural integrity of the individual operations. The thresholds' range considered in the optimized scheduling model is assumed to be identical as presented in Table 7. These parameter ranges are identical to those defined for both the synthetic scenarios and the underlying scheduling model. Mechanistically, lower threshold values trigger the successor activity at an earlier checkpoint, thereby increasing the network's overall degree of parallelism. However, to prevent physically unfeasible or hazardous operational overlap, the optimization space is strictly bounded by these specified engineering ranges.

Throughout the optimization process, all foundational full-precedence, operational, and safety constraints established in the deterministic model are strictly maintained. Specifically, the pushback phase (A12) cannot initiate until all prerequisite ground-handling workflows have concluded and the aircraft footprint is cleared. Similarly, passenger boarding bridge removal is completely restricted until the boarding process terminates, and the terminal taxi-out phase (A13) can only commence following the successful execution of pushback. This controlled implementation of partial parallelization provides a rigorous framework to systematically evaluate diverse scheduling configurations based on their final TTT. Consequently, the optimization objective is to isolate specific network configurations that preserve the core precedence structure and safety boundaries while minimizing idle waiting times and maximizing overall scheduling efficiency.

4.6 Optimized TTT

The optimized schedule keeps the same activity durations as the baseline model in order to evaluate the impact of partial parallelization independently of variations in activity duration. Therefore, changing the time relationships between specific dependent activities is the only way to achieve the improvement. The cargo unloading-to-loading (A5→A6), threshold stay at 100% for this setup, but the deplaning-to-cleaning (A2→A3), deplaning-to-catering (A2→A4) and cleaning-to-boarding (A3→A10) thresholds are set at 60% and 70% respectively (cf. Table 7).

The results for the optimized model are presented in Figure 2. The corresponding TTT in the optimized model is 73.5 minutes, which represents a reduction of 10.5 minutes in comparison to the Baseline model (cf. Figure 1), equivalent to a relative reduction of 12.5%.

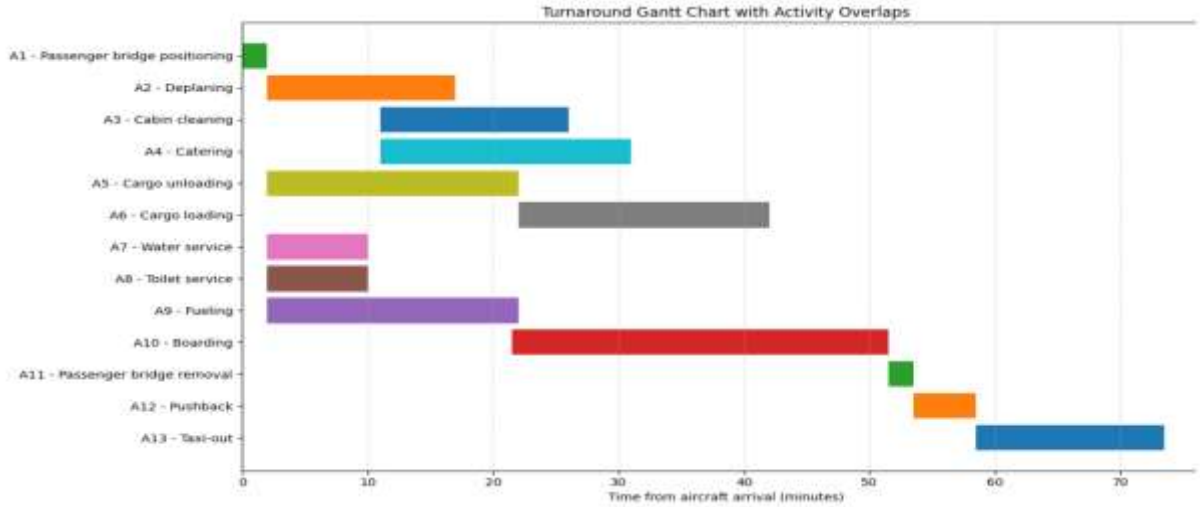


Figure 2 Gantt chart of the Optimized model (cf. Fig. 1)

4.7 Baseline Optimized Timeline and Network Dynamics

By relaxing full-precedence constraints into calibrated completion thresholds, the network achieves significant temporal compression without altering baseline activity durations. The chronological progression of the critical passenger sequence and its key concurrent overlaps.

Following optimization, the definitive critical path retains its foundational topology:

$$A1 \rightarrow A2 \rightarrow A3 \rightarrow A10 \rightarrow A11 \rightarrow A12 \rightarrow A13. \quad (7)$$

The 12.5% reduction in TTT (from 84.0 to 73.5 minutes) is driven entirely by the mathematical compression of non-value-added buffer intervals between sequential tasks rather than shifts in underlying task execution speeds. For instance, applying a 60% completion threshold to the deplaning-to-cleaning (A2→A3) link allows cabin cleaning to initiate at minute 11.0 instead of waiting until minute 17.0. Compounding this, the 70% cleaning-to-boarding (A3→A10) threshold permits boarding to commence at minute 21.5. This combined network elasticity advances the passenger boarding milestone by 10.5 minutes. Passenger boarding (A10) remains the primary operational bottleneck, consuming a dominant 30.0-minute block on the critical path.

Conversely, downstream optimizations applied to non-passenger workflows yield negligible impacts on the final timeline due to the presence of positive slack (float). While lowering the deplaning-to-catering threshold (A2→A4) to 60% accelerates the catering timeline, it fails to compress TTT because catering resides off the critical path and does not bound downstream boarding or pushback gates. Similarly, maintaining a strict 100% threshold on the cargo unloading-to-loading (A5→A6) link results in cargo loading concluding at minute 42.0. Because this sub-sequence terminates well before the resolution of the passenger-boarding phase, farther threshold reductions within the cargo or fueling sectors yield no marginal utility under these nominal task durations. Ultimately, the optimized schedule demonstrates that partial parallelization successfully compresses overall transit windows while preserving the macro-structural integrity of the scheduling network.

4.8 Operational Feasibility and Safety-Adjusted Configurations

While increasing parallel execution across the TA network directly compresses the TTT, a lower mathematical TTT does not guarantee that every concurrent workflow is equally viable under real-world operating conditions. The intersection of distinct ground operations introduces logistical complexities that must be balanced against theoretical efficiency gains.

The most prominent example is the concurrent window between passenger boarding (A10) and cabin cleaning (A3). Under the optimized schedule, cabin cleaning spans from minute 11.0 to 26.0, while passenger boarding initiates at minute 21.5, creating a 4.5-minute physical overlap. Although this parallelization is highly effective at compressing the critical passenger sequence, its practical execution requires stringent gate-side coordination to prevent ramp congestion and maintain service quality.

Additionally, a brief 0.5-minute overlap occurs between fueling (A9) and passenger boarding (A10), as boarding begins at minute 21.5 while fueling concludes at minute 22.0. To mitigate any risks associated with concurrent fueling and passenger embarkation, a more conservative operational configuration can be enforced by adding a strict safety restriction: passenger boarding can only commence after fueling has entirely finished. Implementing this safety mandate delays the initiation of boarding by 0.5 minutes, shifting the final TTT from 73.5 minutes to 74.0 minutes. This safety-adjusted configuration demonstrates that a highly secure, risk-mitigated operational schedule can be achieved while preserving the vast majority of the parallelization-driven time savings.

5 MACHINE LEARNING MODEL

The primary objective of the machine learning pipeline is to expedite the exploration of the multi-dimensional design space and isolate low-TTT operational configurations within the simulated TA environment. As established in the preceding methodology, a RFR is deployed to map the highly complex, non-linear relationships connecting the continuous target variable, TTT, to the underlying vector of 13 activity durations and 4 partial-overlap thresholds.

Crucially, the Random Forest architecture does not natively enforce operational logic, nor does it directly construct the TA schedule. Instead, it functions as a highly scalable meta-model that provides a rapid predictive approximation

of the deterministic network's behavior. By efficiently scanning thousands of parameter permutations, the regressor isolates promising, high-efficiency candidate configurations. These data-driven selections are subsequently re-ingested into the rule-based scheduling framework for exact timeline resolution and constraint validation.

5.1 Synthetic dataset generation

To facilitate supervised ML in the absence of empirical operational data, a robust synthetic dataset consisting of 5000 distinct TA scenarios was constructed. Within this framework, individual activity durations and partial-overlap thresholds were systematically sampled from predefined engineering envelopes.

Rather than executing an unconstrained randomization, the parameter generation process follows strict operational logic. While the 4 overlap parameters are strictly confined to thresholds deemed physically viable by the scheduling framework, the 13 activity durations are sampled from the localized operational intervals established for each TA task. To capture the full spectrum of ground-handling variability, a continuous uniform random distribution is applied within these minimum and maximum boundaries. This stochastic sampling technique introduces simultaneous variance across both individual task execution speeds and the potential degree of parallel execution between dependent workflows.

Table 10 Inputs for the synthetic dataset

Variables	Min durations (minutes)	Max durations (minutes)
Passenger bridge positioning	1	3
Deplaning	10	30
Cabin cleaning	10	20
Catering	10	30
Cargo unloading	10	30
Cargo loading	10	30
Water service	5	10
Toilet service	5	10
Fueling	15	30
Boarding	15	40
Passenger bridge removal	1	2
Pushback	3	5
Taxi-out	10	25
Deplaning to cleaning threshold	60%	100%
Deplaning to catering threshold	60%	100%
Cargo unloading to cargo loading threshold	50%	100%
Cleaning to Boarding threshold	70%	100%

For every parameter combination, the deterministic scheduling model executes its embedded precedence and partial-overlap logic to resolve exact network timelines, determining the definitive start and finish milestones for each operation. Following schedule resolution, downstream operational metrics, specifically the absolute TTT, critical path composition, bottleneck task ID, and bottleneck activity name, are recorded and co-indexed with their corresponding inputs. Consequently, the completed dataset comprises 5000 rows across 22 operational variables. Structurally, each discrete TA profile is characterized by one unique scenario identifier, 13 independent activity durations, and four partial-overlap thresholds, while the remaining 4 variables function as network-derived scheduling outputs. Table 11 illustrates one example of a generated synthetic TA scenario.

Table 11 Example of a generated synthetic TA scenario

Variable	Value
scenario_id	1
bridge_position_duration	2.28
deplaning_duration	10.5
cleaning_duration	12.75
catering_duration	14.46
cargo_unloading_duration	24.73
Cargo_loading_duration	23.53
water_service_duration	9.46
Toilet_service_duration	5.43
fueling_duration	19.22
Boarding_duration	15.74
bridge_removal_duration	1.22
pushback_duration	4.01
taxi_out_duration	10.4
deplaning_to_cleaning_threshold	0.68
deplaning_to_catering_threshold	0.86
cargo_unloading_to_loading_threshold	0.77
cleaning_to_boarding_threshold	0.77
TTT	59.32
bottleneck_task	A6
bottleneck_name	Cargo loading
Critical_path	A1 → A5 → A6 → A12 → A13

5.2 Analysis of the dataset

To evaluate the operational breadth of the design space and verify the structural integrity of the simulation framework, the statistical distribution of the target variable, TTT, was comprehensively evaluated and is illustrated in Figure 3. This analysis confirms that the 5000 synthetic TA scenarios generate a sufficiently robust and heterogeneous distribution of operating conditions.

The generated schedules yield an absolute operational range bounded by a minimum TTT_{min}=53.38 minutes and a maximum TTT_{max}=111.38 minutes. The target distribution exhibits a high degree of symmetry, balancing evenly around a central tendency with a sample mean TTT_{mean}=81.34 minutes and a median of TTT_{median}=81.26 minutes. This near-equivalence between the mean and median indicates a standard normal-like distribution, free from severe skewness toward either highly compressed or excessively delayed operational extremes.

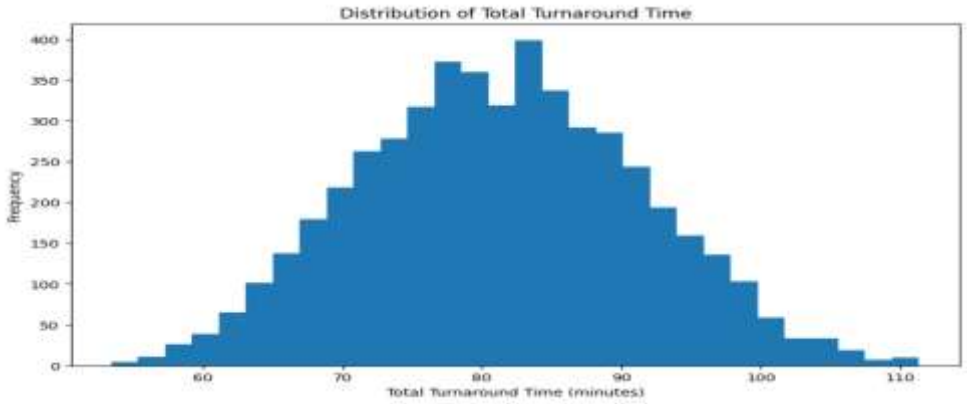


Figure 3 Distribution of TTT in the synthetic dataset (TTT_{min}=53.38 min, TTT_{max}=111.38 min, TTT_{mean}=81.34 min, TTT_{median}=81.26 min)

,)

Furthermore,
the continuous dispersion across the histogram confirms that the stochastic sampling methodology successfully maps a comprehensive spectrum of TA timelines rather than clustering artificially around a singular nominal value. A calculated standard deviation of 10.05 minutes statistically validates this substantial operational variance, confirming that the dataset contains the necessary behavioral diversity to effectively train and test the predictive ML pipeline.

5.3 Analysis of overlap thresholds

Since they establish when dependent activity can begin before its predecessor has finished, the partial overlap thresholds are very important. As a result, they have a impact on the TA process's level of parallel execution and may have an impact on the final TTT. Because of this, the relationship between TTT and each of the 4 overlap criteria (Table 7) was examined in each of the 5000 artificial scenarios.

As the threshold increases, there is a general trend toward higher TA times. Lower threshold values increase the degree of overlap between the two tasks by enabling cabin cleaning to start earlier in the deplaning process (deplaning-to-cleaning, A2→A3) (Figure 4(a)). Nevertheless, there is still a significant range of TTT values for each threshold value, suggesting that the TTT is not solely determined by this parameter.

The relationship between TTT and the deplaning-to-catering (A2→A4) threshold (Figure 4(b)) is significantly weaker. There is no noticeable rise or fall in TTT, and the scenarios are still broadly scattered along the whole threshold range. This implies that the overlap between deplaning to cabin cleaning has a greater direct impact on the final TTT under the simulated conditions than permitting catering to start earlier during deplaning.

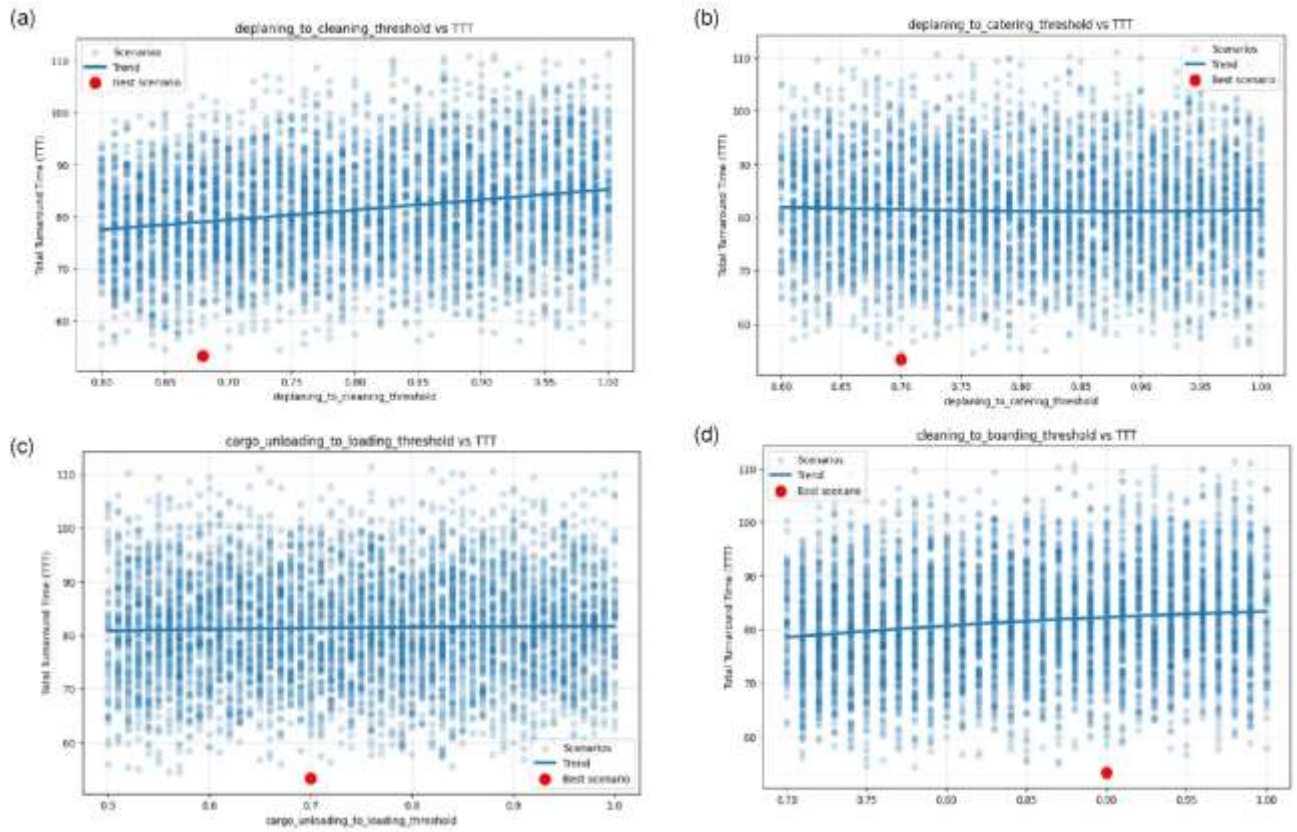


Figure 4 TTT depending on (a) Deplaning-to-cleaning threshold (A2→A3), (b) Deplaning-to-catering threshold (A2→A4), (c) Cargo unloading-to-loading threshold (A5→A6), and (d) Cleaning-to-boarding threshold (A3→A10)

The cargo-unloading-to-loading (A5→A6) threshold (Figure 4(c)) has a similar behavior. TTT readings show no significant pattern and are dispersed throughout the threshold's whole range. This suggests that changes in this overlap condition typically do not affect the TA's final completion time, while cargo operations may still be important in some situations based on how long the related tasks take.

On the other side, there is a stronger correlation between TTT and the cleaning-to-boarding (A3→A10) threshold (Figure 4(d)). In general, lower threshold values enable earlier boarding, which increases overlap in the passenger-handling procedure. However, the observations' dispersion is still noteworthy. Specifically, even though the model allows lower values, the dataset's best-performing scenario has a cleaning-to-boarding threshold of 0.90. This demonstrates that the value of a single overlap parameter cannot be the single cause of a low TTT.

The best-performing scenario from the entire synthetic dataset is highlighted in each graphic. With threshold values of 0.68 for deplaning-to-cleaning (A2→A3), 0.70 for deplaning-to-catering (A2→A4), 0.70 for cargo-unloading-to-loading (A5→A6), and 0.90 for cleaning-to-boarding (A3→A10), this scenario achieves a TTT of 53.38 minutes, corresponding in the figures with a red dot. It is not appropriate to consider these values as ideal thresholds on their own. Rather, they correspond with a specific set of overlap circumstances and activity durations that generated the lowest TTT out of 5000 created scenarios.

Overall, the analysis shows that compared to the catering and cargo overlap thresholds, the passenger-related overlap conditions specifically, deplaning-to-cleaning A2→A3 and cleaning-to-boarding (A3→A10) have a stronger correlation with TA performance. Simultaneously, the wide dispersion seen in all four figures attests to the fact that

TTT is not the product of a single threshold taken into consideration separately, but rather of the simultaneous interplay of several operational variables.

5.4 Feature Selection and Model training

The synthetic dataset contains 22 variables, but only 17 are utilized as predictors in the Random Forest model. The input feature set consists of the 13 activity durations and the 4 partial-overlap criteria, which collectively define the operational configuration of each TA scenario.

TTT serves as the target variable, while the remaining 4 variables are excluded from the ML model for the following reasons:

- scenario_id: Omitted because it is a unique identifier lacking operational information.
- bottleneck_task, bottleneck_name, and critical_path: Omitted because they are downstream outputs generated by the scheduling model. Including them as predictors would cause data leakage.

The Random Forest model was trained on the synthetic dataset using an 80:20 train-test split. Out of the 5000 generated scenarios, 4000 (80%) were allocated to the training set, while the remaining 1000 (20%) were reserved for the test subset to evaluate the model's generalization performance on unseen situations.

5.5 Model Performance

5.5.1 Predictive Performance

The Random Forest model achieved a coefficient of determination (R^2) of 0.949, a Mean Absolute Error (MAE) of 1.85 minutes, and a Root Mean Squared Error (RMSE) of 2.37 minutes.

Because MAE and RMSE measure the variance between predicted and reference TTT values, lower scores indicate superior predictive accuracy. The MAE demonstrates that, on average, the predicted TTT deviates from the scheduling model's baseline by approximately 1.85 minutes. The RMSE is slightly higher at 2.37 minutes because it penalizes larger prediction errors more heavily. The narrow gap between the MAE and RMSE indicates that the test set is not heavily influenced by outlier prediction errors.

An R^2 value closer to 1 states for a stronger alignment between predicted and actual values. The model's score of 0.949 means it accounts for 94.9% of the variation in TTT across all test cases. This combination of a high R^2 and low error metrics proves that the Random Forest model successfully captured the relationships between the operational inputs and the target variable. Consequently, these predictions provide a sufficiently accurate foundation for the subsequent analytical phases of the project.

5.5.2 Feature Importance

Analyzing feature importance (Figure 5) highlights which operational variables exert the greatest influence on the ML model. The data indicates that `boarding_duration` is the most significant variable, yielding an importance score of 0.472. It is followed by `deplaning_duration` (0.201) and `taxi_out_duration` (0.176). While their impact is substantially lower than the top three predictors, `cleaning_duration` (0.051) and `deplaning_to_cleaning_threshold` (0.049) still offer noticeable contributions.

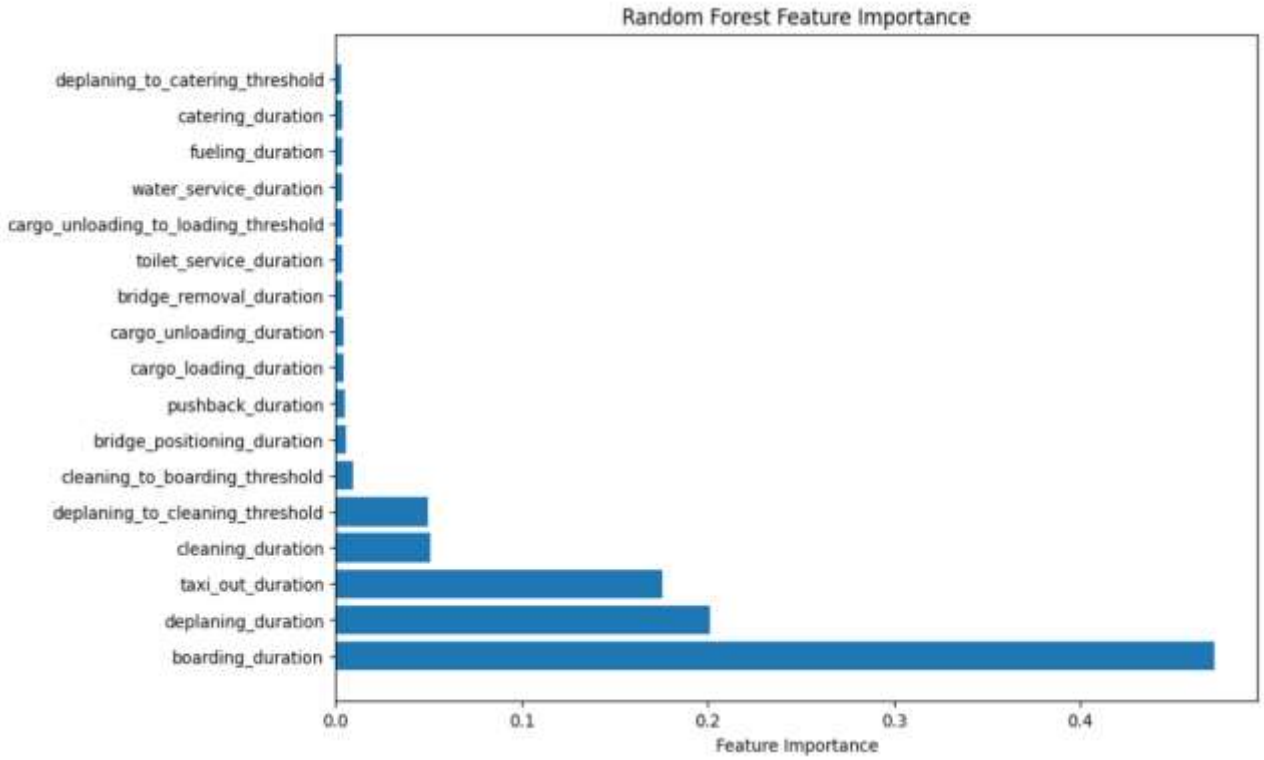


Figure 5 Importance of each feature to Random Forest

This hierarchy strongly reflects the TA architecture described in earlier sections. The critical path is heavily dictated by the passenger-handling sequence, which encompasses boarding, deplaning, and cleaning. Taxi-out also carries substantial weight because it occurs sequentially after pushback, meaning its entire duration directly extends the final TTT calculated by the computational model. Finally, the emergence of the deplaning-to-cleaning (A2→A3) threshold as the highest-ranking overlap parameter aligns consistently with the study's threshold sensitivity analysis.

5.6 Machine Learning Results

5.6.1 Selected Turnaround Configuration

Following model evaluation, the Random Forest regressor was deployed to predict TTT for the unseen test scenarios. The model identified Scenario 928 as the most efficient configuration, featuring an initial deterministic TTT of 54.55 minutes. By applying the deterministic scheduling model to refine the 4 overlap criteria while keeping the 13 original activity durations intact, the final TTT was successfully reduced to 53.01 minutes.

The optimized overlap thresholds for Scenario 928 are defined as follows:

- Cargo Operations: Cargo loading begins after 50% of cargo unloading is complete.
- Cabin Preparation: Cabin cleaning and catering commence after 60% of deplaning is finished.
- Passenger Boarding: Boarding starts after 70% of cabin cleaning is completed.

The 13 activity durations in this configuration listed in Table 12 represent a high-performing combination drawn from the randomized operational ranges established before (cf. Table 10).

Table 12 ML Model parameters (Scenario 928)

Project ID	Activity	Duration

A1	Passenger bridge positioning	1.3
A2	Deplaning	13.04
A3	Cabin cleaning	12.07
A4	Catering	12.36
A5	Cargo unloading	23.74
A6	Cargo loading	24.01
A7	Water service	7.3
A8	Toilet service	7.33
A9	Fueling	21.6
A10	Boarding	16.34
A11	Passenger bridge removal	1.61
A12	Pushback	4.49
A13	Taxi-out	11.34

5.6.2 Critical Path and Optimization Analysis

Evaluating the configuration presented in Table 12 in the deterministic scheduling model yields a TTT of 53.01 minutes. The cargo loading (A6) activity acts as the bottleneck, establishing a critical path of

$$A1 \rightarrow A5 \rightarrow A6 \rightarrow A12 \rightarrow A13. \quad (8)$$

Notably, the individual activity durations of this configuration do not match the minimum boundaries defined for the synthetic dataset. Because the 5000 randomized scenarios represent only a small sample of all potential operational combinations, this selection reflects a strong low-TTT solution found within the generated dataset rather than a theoretical global minimum.

For instance, several activities in Scenario 928 exceed their absolute minimum thresholds:

- Taxi-out (A13): Takes 11.34 minutes (minimum possible: 10.0 minutes)
- Passenger bridge positioning (A2): Takes 1.3 minutes (minimum possible: 1.0 minute)

Reducing these specific tasks to their minimum limits while maintaining the same overlap thresholds would yield an even lower TTT. Consequently, this configuration serves as a robust baseline for localized optimization rather than the absolute theoretical limit of the system.

5.6.3 Resulting Schedule for ML model

The ML-assisted schedule achieves a TTT of 53.01 minutes, characterized by a high degree of parallel execution (Figure 6). Following passenger bridge positioning (A1), deplaning (A2), and cargo unloading (A5) begins at minute 1.30. Triggered by the 60% deplaning threshold, cabin cleaning (A3) and catering (A4) both commence at minute 9.12. This core TA phase enables the parallel execution of fueling (A9), water service (A7), lavatory servicing (A8), and cargo loading (A6).

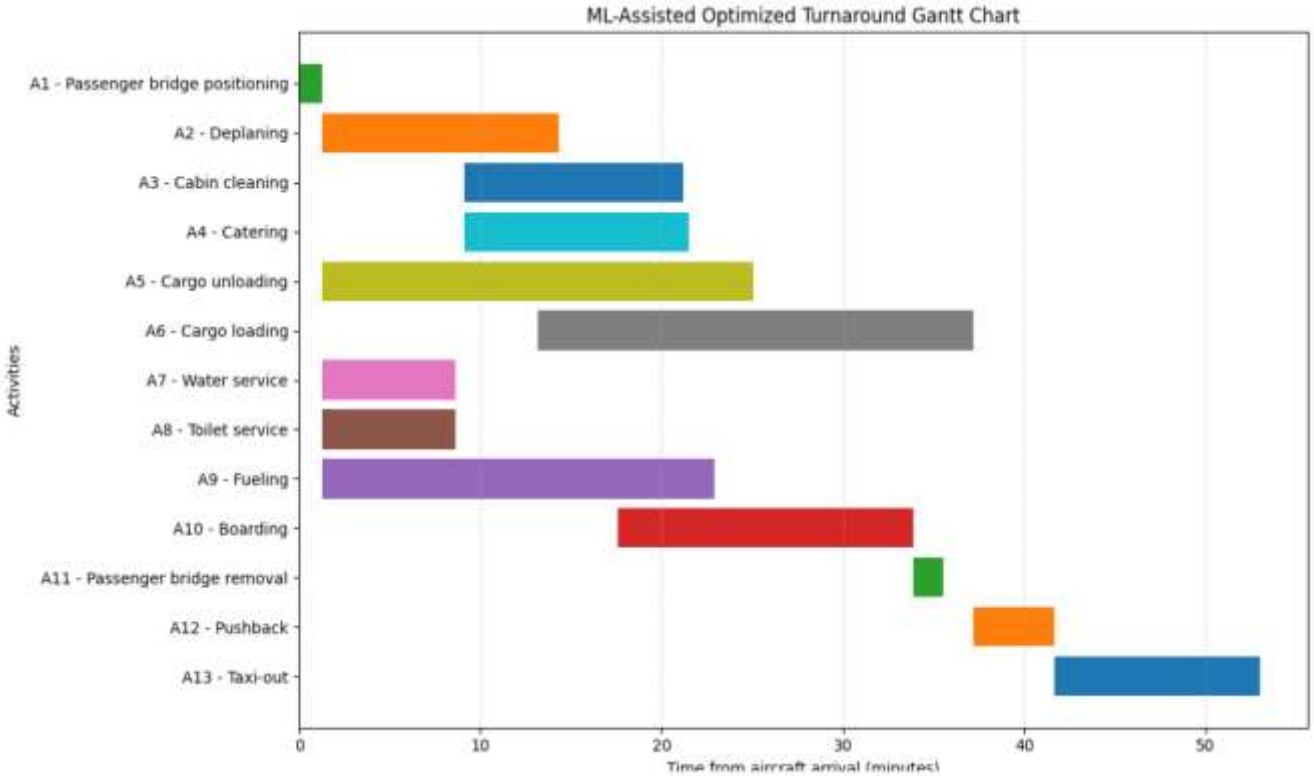


Figure 6 Gantt chart of the schedule selected by ML

The key operational milestones for the ML model proceed as follows:

- Cargo Loading (A6): Starts at minute 13.17 (upon 50% cargo unloading) and concludes at minute 37.18, acting as the primary constraint on the pushback schedule.
- Passenger Boarding (A10): Begins at minute 17.57 after 70% of the cabin cleaning is completed.
- Departure Phase: Pushback (A12) concludes at minute 41.67, followed by an 11.34 minute taxi-out phase (A13) to reach the final 53.01 minute TTT.

The corresponding critical path is validated as:

$$A1 \rightarrow A5 \rightarrow A6 \rightarrow A12 \rightarrow A13 \quad (9)$$

5.6.4 Safety-Adjusted Schedule

In the initial ML-assisted configuration, passenger boarding (A10) and fueling (A9) overlapped for approximately 5.33 minutes, as boarding began at minute 17.57 while fueling concluded at minute 22.90.

To enforce a more conservative operational profile, a safety restriction was introduced requiring fueling to be completely finished before passenger boarding could commence. Consequently, the start of boarding was deferred from minute 17.57 to minute 22.90. This adjustment delayed the subsequent sequence, pushing the completion of boarding to minute 39.24.

The results of this safety modification are illustrated in the Gantt shart in Figure 7.

- TTT increased by 3.67 minutes, rising from 53.01 minutes to 56.68 minutes.
- The critical path is shifted. The bottleneck shifted from the cargo sequence to the passenger/fueling sequence, changing from A1 → A5 → A6 → A12 → A13 to the new critical path

$$A1 \rightarrow A9 \rightarrow A10 \rightarrow A11 \rightarrow A12 \rightarrow A13. \quad (10)$$

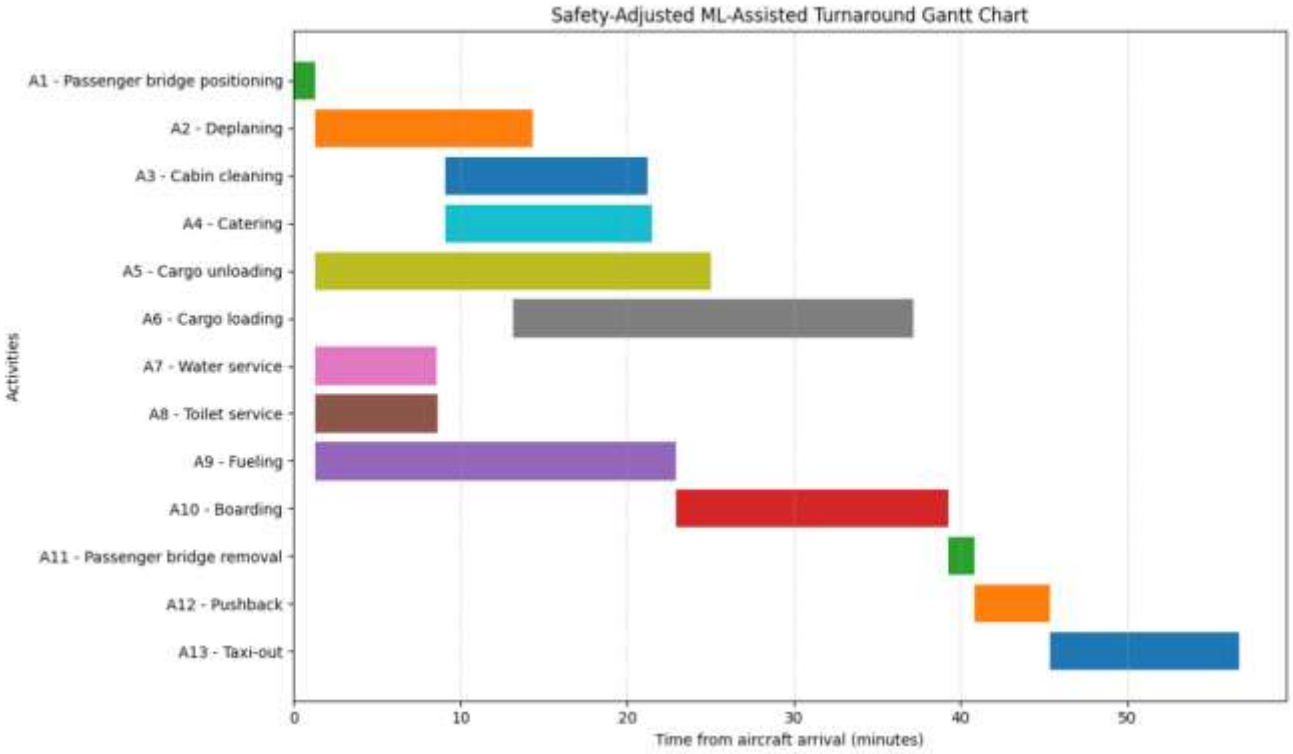


Figure 7 Gantt chart of the safety-adjusted schedule selected by ML

6 COMPARISON AND ANALYSIS: ML-ASSISTED VS BASELINE OPTIMIZED SCHEDULE

6.1 Performance and Duration Profiles

Under identical safety constraints, the ML-assisted schedule achieves a TTT of 56.68 minutes, while the Baseline Optimized schedule reaches 74.0 minutes (Table 13). This represents a reduction of 17.32 minutes, or approximately 23.4%. This performance gap stems from how each approach handles activity durations. The Baseline Optimized model deliberately holds the baseline task durations constant to isolate the impact of parallelization. Conversely, the ML-assisted approach evaluates a broader range of operational setups by varying both activity durations and overlap thresholds across the synthetic dataset.

The resulting configuration reveals that the ML model does not simply minimize every task duration across the board:

- Extended Durations: Cargo loading, cargo unloading, and fueling actually take longer in the ML-assisted configuration than in the Baseline Optimized schedule.
- Compressed Durations: Key passenger and departure operations are significantly shorter. Most notably, boarding drops from 30.00 to 16.34 minutes, and taxi-out is reduced from 15.00 to 11.34 minutes.

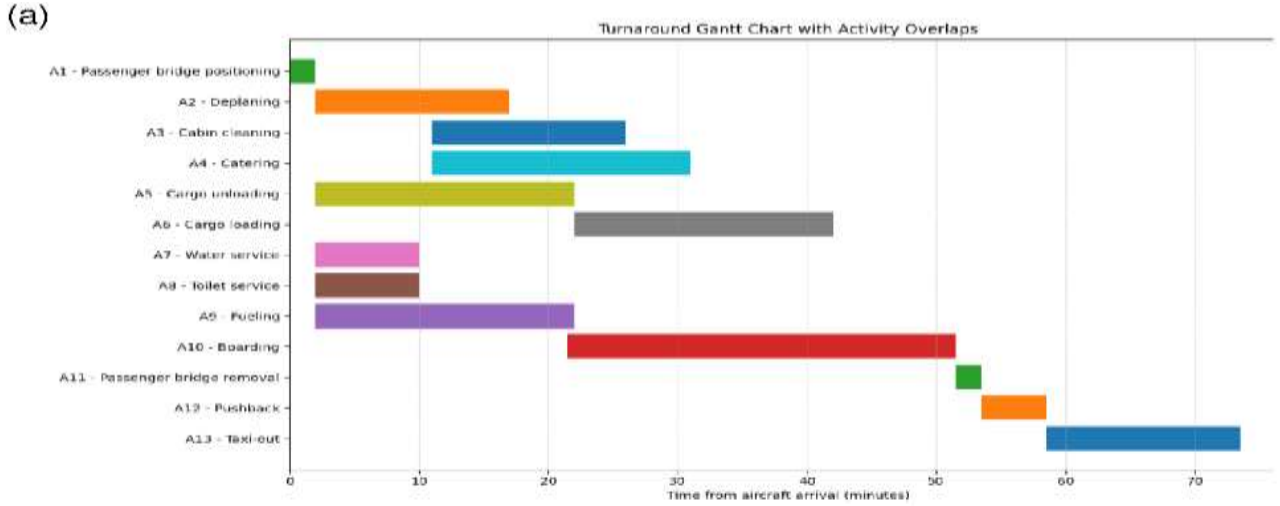
This distribution aligns with the Random Forest feature importance analysis, which identified boarding, deplaning, and taxi-out as primary drivers of TTT. By favoring shorter durations for high-impact critical variables, the model achieves a net time savings despite letting lower-priority activities run longer.

Table 13 Comparison Baseline Model vs ML Model parameters (cf. Tables 9 and 12)

Project ID	Activity	Optimized Baseline	ML- Assisted	Difference (min)

		Duration (min)	Dduration (min)	
A1	Passenger bridge positioning	2	1.3	-0.7
A2	Deplaning	15	13.04	-1.96
A3	Cabin cleaning	15	12.07	-2.93
A4	Catering	20	12.36	-7.64
A5	Cargo unloading	20	23.74	+3.74
A6	Cargo loading	20	24.01	+4.01
A7	Water service	8	7.3	-0.7
A8	Toilet service	8	7.33	-0.67
A9	Fueling	20	21.6	+1.6
A10	Boarding	30	16.34	-13.66
A11	Passenger bridge removal	2	1.61	-0.39
A12	Pushback	5	4.49	-0.51
A13	Taxi-out	15	11.34	-3.66
A1-A13	TTT	74	56.68	-17.32

(a)



(b)

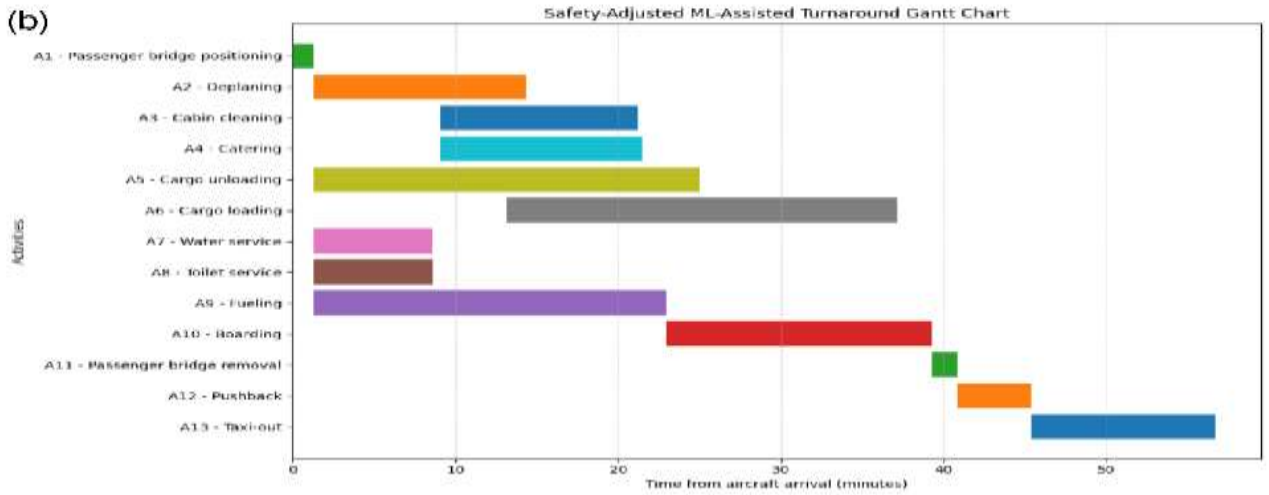


Figure 8 Gantt charts: comparison between (a) the Baseline Optimized model and (b) the safety-adjusted ML model

6.2 Network Mechanics and Parallelization Logic

Because bottleneck and critical path outputs were excluded from training to prevent data leakage, the Random Forest model never explicitly computes the critical path. Instead, it implicitly learns how the 13 activity durations and 4 overlap thresholds interact to determine the final TTT. This allows the model to optimize the holistic temporal configuration rather than treating individual task lengths in isolation.

Figure 8 shows a comparison between the Baseline Optimized model and the safety-adjusted ML model. Out of the four parameters, three converged on identical values in both models: Deplaning-to-Cleaning (A2→A3): 60%, Deplaning-to-Catering (A2→A4): 60%, Cleaning-to-Boarding (A3→A10): 70%. The alignment across these passenger-handling thresholds indicates that the machine learning analysis largely validates the scheduling logic built into the Baseline Optimized model. The primary divergence occurs in the cargo unloading-to-loading threshold (A5→A6), which drops from 100% (strictly sequential) in the Baseline Optimized model to 50% (partially parallel) in the ML-assisted configuration. This increased parallelization compensates for the prolonged cargo-handling durations observed in the ML scenario.

Ultimately, the ML findings reinforce rather than replace the deterministic scheduling model. While deterministic optimization successfully identifies ideal thresholds for a single baseline scenario, the ML model confirms that these parallelization strategies remain valid across a vast operational design space.

6.3 Robustness Analysis

A campain with 1000 MCSs evaluated the resilience of the final safety-adjusted schedule by applying a $\pm 10\%$ independent variance to individual activity durations while holding the optimized overlap thresholds and fueling safety constraints constant. Compared to the deterministic baseline TTT of 56.68 minutes, the simulation yielded the following stability metrics:

- TTT_{mean}: 56.73 minutes (with a standard deviation of 1.67 minutes).
- TTT_{range}: 52.00 to 60.94 minutes across all iterations.
- 95th PT: 59.47 minutes, meaning 95% of simulated cases remained within 2.79 minutes of the baseline.

The minimal variance between the baseline and the simulated mean demonstrates that the schedule avoids compounding delays under minor operational fluctuations. This confirms that the time-performance advantages are resilient and not uniquely dependent on the exact durations of Scenario 928. However, this analysis strictly models task execution uncertainty and excludes broader operational disruptions, which are detailed as limitations in the following section.

6.4 Main Limitations

The primary limitation of this study is the reliance on a synthetic dataset rather than actual operational TA data. However, worth pointing out that this work presents a feasibility study instead a concrete case study on any particular airport. While the 5000 randomized configurations expose the Random Forest model to a broad combination space, the learned relationships remain restricted by the predefined activity ranges, precedence logic, and scheduling assumptions established in this project.

Additionally, the scheduling model simplifies real-world airport operations. By focusing on 13 activities and 4 partial-overlap thresholds, it omits critical resource constraints such as staff availability, ground-support equipment conflicts, stand congestion, and vehicle interference. Safety protocols are also idealized, whereas actual airport deployments require complex, facility-specific constraints.

Finally, the robustness analysis only accounts for a $\pm 10\%$ variance in task durations. While this demonstrates

resilience under minor uncertainty, it does not replicate major operational disruptions, cascading delays, or unexpected system shocks. Consequently, these findings should be interpreted strictly as an evaluation of the proposed approach within a controlled environment.

6.5 Environmental perspective

Since this project mostly relies on computational work using Python, synthetic data creation, and ML and does not require physical prototypes or additional airport infrastructure, the direct environmental effect connected with its development is relatively low. If shorter ground operations minimize the consumption of aircraft engines or ground-support equipment, increasing TA efficiency might have a favorable impact on the environment. According to Nikoleris et al. [22], taxi operations are especially important because airplanes continue to use fuel and emit emissions when they are on the ground. However, since fuel consumption and emissions are not directly calculated and considered in this study, a lower TTT should not be automatically taken as a benefit in the environment. Additionally, if increased TA efficiency allows for more flights and expands airport capacity, the ensuing increase in traffic may partially or entirely outweigh the environmental gains obtained per operation. As a result, a more comprehensive analysis encompassing emissions, fuel consumption, and traffic demand would be necessary to determine the overall environmental impact.

7. SUMMARY AND CONCLUSION

This study demonstrates that combining deterministic scheduling, partial parallelization, and ML can significantly reduce aircraft TTT. Our findings reveal that TA performance depends not only on isolated task durations but critically on precedence logic, parallel execution capacity, and network placement.

Initially, deterministic analysis proved that moderate parallelization yields substantial temporal compressions without accelerating individual tasks; the Baseline Optimized plan reduced TTT by 12.5% (from 84 to 73.5 minutes) using identical task durations. To explore broader operational environments, an ML pipeline was trained on a synthetic dataset of 5000 scenarios. The resulting RFR achieved high predictive accuracy (R^2 = 0.949), MAE = 1.85 minutes), identifying an unseen high-efficiency network . When subjected to the strict safety constraint mandating that boarding only begin after fueling concludes, the final ML-assisted configuration achieved a TTT of 56.68 minutes, a 23.4% (17.32 minute) improvement over the safety-adjusted baseline.

The ML-assisted analysis largely reinforced the baseline scheduling logic, retaining 3 out of 4 optimal overlap thresholds. It highlighted that an activity's operational criticality is dictated by its interaction with simultaneous tasks along the critical path rather than its absolute duration. Finally, a robustness analysis using 1000 MCSs ($\pm$10% task variance) confirmed schedule stability, yielding a mean TTT of 56.73 minutes and a P95 of 59.47 minutes. Ultimately, this hybrid framework serves as a valuable predictive approximation of operational variability, though validation with live empirical data remains a necessary prerequisite for industry deployment.

Future Work

Future research should prioritize validating these models using empirical TA data. Consider real world operational records will allow for the calibration of activity ranges and dependency rules, ensuring the ML model reflects genuine airport environments. Aside, the deterministic scheduling model can be expanded by integrating granular operational constraints, such as staff availability, ground-support equipment allocation, stand congestion, and vehicle taxi interference. Additionally, developing a dynamic decision-support framework would enable real-time schedule updates, allowing operators to automatically recalculate the predicted TTT when live execution times drift from baseline estimates. Finally, the ML architecture can be refined through rigorous performance benchmarking. Evaluating alternative regression algorithms, implementing k-fold cross-validation [23], and testing stability across multiple random training seeds will ensure the predictive models remain robust and highly generalized before actual deployment.

ACKNOWLEDGMENT
S. A. is a Serra Húnter Fellow. This work was supported by the Spanish Government under grants PID2023-150014OB-C21 and PID2023-150029NB-I00.

DECLARATION OF INTEREST
The authors have no conflicts of interest.

DATA AVAILABILITY
The data that support the findings of this study will be made available by the corresponding author on reasonable request.

REFERENCES
[1] Schmidt, M. (2017) A review of aircraft turnaround operations and simulations. Prog. Aerosp. Sci., 92, 25-38, 10.1016/j.paerosci.2017.05.002.
[2] Mota, M. M. Boosten, G. De Bock, N. Jimenez, E. and de Sousa, J. P. (2017). Simulation-based turnaround evaluation for Lelystad Airport, Journal of Air Transport Management, 64, 21-32, 10.1016/j.jairtraman.2017.06.021.
[3] Schultz, M. Kunze, T. Oreschko, B. and Fricke, H. (2013). Microscopic process modelling for efficient aircraft turnaround management, Proceedings of the International Air Transport and Operations Symposium (ATOS), Toulouse.
[4] Schmidt, M. (2017). A review of aircraft turnaround operations and simulations, Progress in Aerospace Sciences, 92, 25-38, https://doi.org/10.1016/j.paerosci.2017.05.002.
[5] Norin, A. Granberg, T. A. Yuan, D. and Värbrand, P. (2012). Airport logistics - A case study of the turn-around process, Journal of Air Transport Management, 20, 31-34, https://doi.org/10.1016/j.jairtraman.2011.10.008.
[6] Schultz, M. (2018). Fast aircraft turnaround enabled by reliable passenger boarding, Aerospace, 5(1), 10.3390/aerospace5010008.
[7] Neumann, S. (2019). Is the boarding process on the critical path of the airplane turnaround? European Journal of Operational Research, 277(1), 128-137, 10.1016/j.ejor.2019.02.001.
[8]Wu, Cheng-Lung. (2007). Monitoring Aircraft Turnaround Operations -- Framework Development, Application and Implications for Airline Operations, Transportation Planning and Technology, Taylor & Francis Journals, 31(2), 215-228.
[9] Caswell, T, Story, K. and Frongillo, R. (2007). The Airplane Seating Problem. Mathematical Contest in Modeling, Consortium for Mathematics and Its Applications (COMAP).
[10] Bauer, M. Bhawalkar, K. Edwards, M. (2007). Boarding at the Speed of Flight. Mathematical Contest in Modeling, Consortium for Mathematics and Its Applications (COMAP).
[11] Bachmat, Elkin, 2006 Bounds on the Performance of back-to-front airplane boarding policies,
[12] Fricke, H. Lehmann, S. (2006) Cabin Operation – Service Handling within Final Status Report within Phase II, "Cabin Operation – Service Handling" within the project "Cabin Architecture and Design Concept" (CaaDC).
[13] Airbus Industrie, 1995 A320, Airplane Characteristics For Airport Planning, AC, Chapter 5 Terminal Operation.
[14] Picchi Scardaoni, M. Magnacca, F. Massai, A. and Cipolla, V. (2021) Aircraft turnaround time estimation in early design phases: Simulation tools development and application to the case of box-wing architecture. J. Air Transp. Manag., 96, 102122, doi:10.1016/j.jairtraman.2021.10212
[15] International Civil Aviation Organization. (2019). Manual on ground handling (1st ed.). Doc 10121. Montréal, Québec, Canada, url:https://www.bazl.admin.ch/dam/de/sd-web/T8RbkC5TagYw/icao-doc_10121_manual_on_ground_handling.pdf
[16] Sheibani, K. (2020) Scheduling Aircraft Ground Handling Operations Under Uncertainty Using

Critical Path Analysis and Monte Carlo Simulation: Survey and Research Directions. Int. J. Bus. Strategy Autom., 1(1), 37-45, doi: 10.4018/IJBSA.2020010103.
[17] Padrón S. and Guimarans, D. (2019) An Improved Method for Scheduling Aircraft Ground Handling Operations From a Global Perspective. Asia-Pac. J. Oper. Res., 36(4), 1950020, doi: 10.1142/S0217595919500209.
[18] Sahadevan, D.Ali, H. Al Notman, D. and Mukandavire, Z. (2023) Optimising Airport Ground Resource Allocation for Multiple Aircraft Using Machine Learning-Based Arrival Time Prediction. Aerospace, 10(6), 509, doi: 10.3390/aerospace10060509.
[19] Tang, X. Wu, J. Wu, C.-L. Ding, Y. and Zhang, S. (2025) Dynamic prediction of aircraft turnaround milestone times using a cascaded gradient boosting model for improved airport collaborative decision-making. J. Air Transp. Manag., vol. 128, p. 102842, doi:10.1016/j.jairtraman.2025.102842.
[20] Volt, J. Had, P. Stojić, S. and Delahaye, D. (2026) Improving aircraft ground handling times prediction using machine learning approaches, Transp. Policy, 184, 104196, doi:10.1016/j.tranpol.2026.104196.
[21] Simaiakis I. and Balakrishnan, H. (2010) Impact of Congestion on Taxi Times, Fuel Burn, and Emissions at Major Airports. Transp. Res. Rec. J. Transp. Res. Board, 2184(1), 22-30, doi: 10.3141/2184-03.
[22] Nikoleris, T. Gupta, G. and Kistler, M. (2011) Detailed estimation of fuel consumption and emissions during aircraft taxi operations at Dallas/Fort Worth International Airport. Transp. Res. Part Transp. Environ., 16(4), 302-308, doi: 10.1016/j.trd.2011.01.007.
[23] Gorriz, J.M., Martín Clemente, R., Segovia, F., Ramírez, J., Ortiz, A. y Suckling, J. (2026). Is K-fold cross validation the best model selection method for machine learning? Information Fusion, 135, 104404. https://doi.org/10.1016/j.inffus.2026.104404