Dynamically Predicting Airliner's Turnaround Time

Camille Derie

Dynamically Predicting Airliner’s Turnaround Time

Thesis report

by

Camille Derie

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on August 6, 2025 at 13:00

Thesis committee:

Chair:	Dr.ir. M.I. Gerritsma
Academic Supervisors:	Dr. M.J. Ribeiro
	Dr. N.A.K. Doan
Industry Supervisors:	P. Keusch
	L. Caranti
External examiner:	Dr. O.A. Sharpans’kykh
Place:	Faculty of Aerospace Engineering, Delft
Project Duration:	January, 2025 - July, 2025
Student number:	5009197

Faculty of Aerospace Engineering · Delft University of Technology

Preface

This thesis marks the culmination of my Master of Science in Aerospace Engineering at Delft University of Technology. The research presented was conducted in collaboration with an international European airline, as part of a broader initiative to improve the reliability and efficiency of airline operations. The project explores the prediction of aircraft turnaround time under operational uncertainty, aiming to deliver probabilistic forecasts that adapt dynamically to real-time developments. Drawing on supervised machine learning and multi-source airline data, the study evaluates whether such models can provide operationally meaningful insight across a broad prediction horizon. The work extends both academic understanding and practical tools for day-of-operations decision support.

I would like to express my sincere gratitude to my academic supervisors at TU Delft, Dr. Marta Ribeiro and Dr. Anh Khoa Doan, for their guidance, valuable feedback, and support throughout the course of this project. At Swiss International Air Lines, I am especially grateful to my supervisors Patrick Keusch and Leonardo Caranti, whose unlimited support, trust, and friendship made this an exceptional experience. It has been an honour to work with and learn from them.

My thanks also go to the Data Science Delivery Team, where I completed my internship prior to this thesis. Their encouragement and openness were instrumental in making this project possible. I feel fortunate to have been part of such an inspiring group of colleagues and friends.

Lastly, I would like to thank my family and my dearest friends in Delft, especially the BDE crew, for making the past six years as a student unforgettable. I am also deeply grateful to my friends back in Belgium and here in Zurich, whose support and companionship have meant a great deal.

Camille Derie
Zurich, July 2025

Contents

List of Figures

List of Tables

Part I

Literature Review & Research Definition (week 6)

*Main deliverable for the Research Proposal Review meeting

1 General Introduction

Irregular airline operations (IROP) impose significant financial, operational, and environmental costs on carriers worldwide. Flight delays, cancellations, and inefficiencies disrupt networks, reducing passenger satisfaction and costing airlines billions annually, $8.3 billion in direct expenses and $16.7 billion in lost passenger time value in the U.S. alone [1], with global estimates reaching $25 billion [2, 3]. Beyond financial losses, IROPs contribute to higher fuel consumption and CO_2 emissions. For instance, reducing auxiliary power unit (APU) runtime by one minute per flight could cut over 172 million kg of CO_2 annually across 32 million flights [4]. Among the least efficient 25% of airlines, excessive APU use adds over 85 kg of CO_2 per flight [4], highlighting the urgent need for operational optimization.

Aircraft turnaround time (TAT), the critical interval between an aircraft's arrival at a gate ("on-block") and departure ("off-block"), is crucial to airline operational efficiency. In an industry where margins are very small and disruptions cascade exponentially, even minor TAT variability can trigger systemic inefficiencies, financial losses, and environmental harm.

Both airports and airlines are therefore working on solutions to minimise delays caused by TAT variabilty. For instance, Eindhoven Airport's implementation of the Deep Turnaround AI system, which uses apron-mounted cameras to automatically detect ground handling milestones, such as the start and end times of ground handling activities, enabling better prediction of when an aircraft will be ready for departure [5]. Similarly, Ljubljana Jože Pučnik Airport in Slovenia implemented an AI-enabled turnaround solution, across all critical gates, covering 96% of passenger traffic. This deployment led to an average reduction of nearly six minutes in turnaround delays for flights where alerts were acknowledged, compared to those without. The system enhances operational efficiency by providing real-time visibility into turnaround operations, enabling staff to promptly address bottlenecks and reallocate resources as needed [6].

Major hub carriers are particularly vulnerable to irregularities caused by TAT variability because they typically operate using a banked system. In such systems, flights are scheduled to arrive and depart in concentrated "banks", creating peaks in the demand for ground services. Consequently, even a small irregularity can trigger a cascade of delays, as subsequent flights wait for the delayed arrival of an aircraft, crew, passengers, or baggage. In fact, research by Eurocontrol has found that rotational reactionary delays account for as much as 89% of all reactionary delay minutes [7]. A key challenge in managing these delays is the uncertainty surrounding the minimum turnaround time. If airlines had more accurate information about the TAT under different conditions, they could better anticipate potential disruptions, adjust scheduling proactively, and mitigate reactionary delays.

This research explores how a probabilistic, uncertainty-aware machine learning approach can improve dynamic turnaround time prediction, providing airlines with more accurate insights to mitigate cascading delays. By integrating real-time, multi-source data, encompassing ATC schedules, weather conditions, and detailed operational records, this approach aims to capture the inherent variability of turnaround operations and quantify the associated uncertainty. Unlike traditional models that yield static, deterministic estimates, a dynamic prediction framework could continuously update turnaround time estimates throughout the tactical phase, offering actionable insights into the expected turnaround duration and the uncertainty.

2 Problem Definition

This chapter presents the operational and methodological challenges inherent in predicting aircraft turnaround times. Section 2.1 provides a comprehensive explanation of the turnaround process, outlining the sequence of tasks from aircraft arrival to departure. In Section 2.2, current approaches to turnaround time prediction are reviewed, highlighting the limitations of sub-process modelling and the benefits of a holistic probabilistic framework. Finally, Section 2.3 defines the scope of the research by specifying the prediction time points, the focus on a major European hub airline, and the integration of multiple real-time data sources.

2.1. What is a turnaround process?

The aircraft turnaround process covers the period between an aircraft's arrival at the gate (on-block) and its subsequent departure (off-block). This interval, known as turnaround time (TAT), is very important for airline efficiency and profitability, as aircraft generate revenue only while airborne. Consequently, minimizing ground time through a well-orchestrated sequence of tasks is essential, as shown by the simplified flowchart in Figure 2.1. One should note that the taxi time, the time required for an aircraft to travel between the runway and the gate, is not included in the turnaround time.

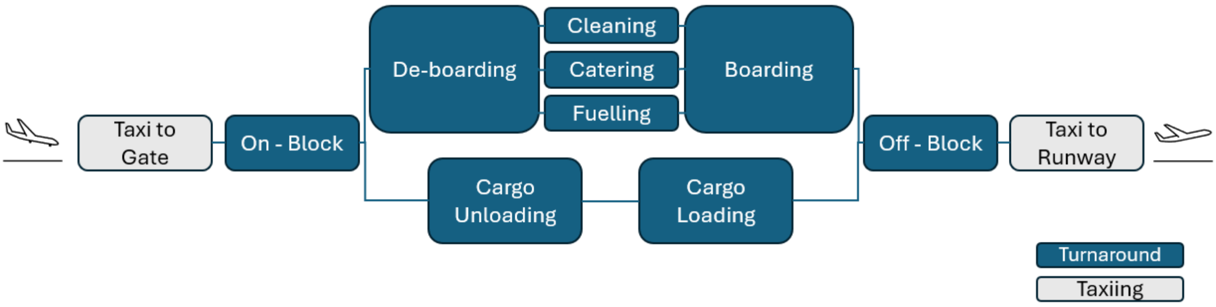


Figure 2.1: Simplified Aircraft Turnaround Process Flowchart (own work)

Upon arrival, ground staff initiate preparations even before the aircraft reaches the gate. Once parked, the aircraft is secured with chocks, and ground power units are connected to maintain onboard systems. Passengers then disembark, allowing crews to commence unloading baggage and cargo. Simultaneously, cleaning teams enter to service the cabin, ensuring it meets hygiene standards for the next set of passengers.

Refuelling operations are conducted concurrently, adhering to strict safety protocols. Catering services restock food and beverages, while maintenance personnel perform routine inspections to identify and address any technical issues. Ground handling teams manage the loading of baggage and cargo for the upcoming flight, coordinating closely to prevent delays.

Passenger boarding is a meticulously timed operation, often initiated before all servicing tasks are complete to expedite departure. Coordination among various teams is crucial as any delay in one area can cascade, affecting the overall schedule. For short-haul flights, a typical turnaround lasts 60 minutes, however, airlines like Ryanair have achieved turnarounds in as little as 25 minutes. Long-haul flights typically require between 90 minutes to two hours due to additional servicing needs. [8]

Given the complexity of these processes, it is vital for airports and airlines to understand the variability and uncertainty connected to these processes, in order to avoid delays, and optimise scheduling of flights, or gate allocations.

2.2. Current Approaches

Considerable research has focused on optimizing and predicting turnaround time, typically by modelling each sub-process through discrete event or agent-based simulations [9, 10, 11]. These methods dissect the turnaround into smaller activities, such as baggage unloading, cabin cleaning, and refuelling, and use data from process mining to identify critical paths and bottlenecks. While this sub-process approach brings insights that can significantly enhance turnaround efficiency, it carries inherent uncertainties: each individual simulation or prediction introduces error margins, which accumulate across multiple sub-processes [12]. This poses a significant challenge, as a turnaround process typically entails more than 10 sub-processes. Furthermore, the sub-process time data that is used to build these models often comes from manual timestamp entries, that could further inflate these error margins. Existing studies have tried quantifying and accounting for this cumulative uncertainty [13, 14]. This uncertainty is also related to the prediction horizon. They rely on many operational factors, so most research focuses on making prediction during the turnaround itself, to avoid high uncertainty.

To mitigate these challenges, some researchers have proposed a more holistic approach to turnaround time prediction [15]. Instead of estimating every sub-process, this method aims to produce a probability distribution of how long the entire turnaround will take. Although this strategy forgoes granular insights into specific bottlenecks, it reduces the compounding uncertainties found in sub-process modelling and can be highly beneficial for improving operational resilience. By delivering a robust estimate, or a probability distribution, of the total turnaround time, such a perspective offers a valuable tool for optimizing schedules and managing disruptions. This research aims to build further on this approach.

2.3. Problem Scope

This research focuses on predicting aircraft turnaround time as a probability distribution rather than a single deterministic estimate, enabling a more comprehensive representation of operational variability. Two key time points have been selected, at which predictions will be most relevant for operations. The first prediction is made immediately after the previous flight in the turnaround takes off, and the second prediction occurs just before the previous flight lands. This dynamic approach ensures that predictions remain relevant as new data becomes available, providing updated estimates that reflect evolving conditions.

The study is specifically scoped to an international European hub airline, with initial predictions limited to turnarounds occurring at its primary hub airport. By focusing on a single airline and airport, the research ensures that the developed model is well-aligned with the operational complexities of a large hub, where tightly coordinated ground handling processes and banked flight schedules make turnaround efficiency critical. While the approach may be extendable to other airports and carriers, the study remains within this controlled environment to ensure feasibility and robust validation.

To support accurate and meaningful predictions, the model will leverage a diverse range of real-world and real-time data sources. These include the airline's own operational records, detailing flight schedules, fleet movements, and ground handling activities, as well as external factors such as weather conditions, air traffic control constraints, and broader airport congestion metrics. By integrating multiple sources of information, the research aims to capture key operational dependencies that influence turnaround time variability.

This study does not focus on optimising the turnaround process itself but rather on improving the predictability of turnaround durations. The objective is to provide airlines with more accurate and uncertainty-aware estimates of turnaround times, enabling better schedule planning and proactive decision-making, particularly in irregular operations. While the research will assess prediction accuracy in an operational setting, aspects such as ground handling process redesign or resource allocation strategies fall outside the scope of this study.

3 Literature Review

Understanding and predicting aircraft turnaround time is a crucial aspect of airline operations, directly impacting scheduling efficiency, delay mitigation, and overall network performance. This chapter reviews existing research on turnaround prediction and optimisation, covering a range of methodologies from traditional analytical techniques to state-of-the-art probabilistic machine learning models.

A comprehensive review of the literature in Section 3.1 synthesises key studies, highlighting the shift from deterministic approaches to data-driven predictive models. Simulation-based methods, including discrete-event simulation (DES) and agent-based simulation (ABS) (Section 3.2), are examined to understand their role in modelling turnaround processes. Given the inherent variability in airport operations, the treatment of uncertainty and its impact on predictive accuracy is discussed in Section 3.3. Additionally, strategies for buffer and slack allocation (Section 3.4) are explored as mechanisms for mitigating delay propagation. Finally, a probabilistic modelling approach is considered in Section 3.5, offering a robust framework for forecasting turnaround times under varying operational conditions.

By identifying key challenges, such as the limitations of deterministic predictions and the complexities of integrating multi-source real-time data, this chapter lays the groundwork for the formulation of research objectives and questions in Chapter 4.

3.1. A Review on Turnaround Research

This section synthesises the literature on aircraft turnaround operations based on the studies summarised in Table 3.1. The table details the methods, prediction horizons, input data, uncertainties, future work recommendations, and key findings of each study, thus providing a unified view of recent advances in the field.

A clear trend in the reviewed literature is the shift from traditional analytical and simulation models to more advanced, data-driven approaches. For example, Cui et al. [11] employ a Time Transition Petri Net combined with Bayesian dynamic updates to achieve real-time predictions. Similarly, Luo et al. [9, 10] utilise machine learning techniques, including decision trees, random forests, and XGBoost, in tandem with fusion logic and agent-based simulation, to capture both sequential dependencies and micro-level interactions in turnaround processes. These developments align with the emphasis in [18], a review article on turnaround research, on incorporating real-world operational data to enhance model performance.

Another recurring theme is the rigorous treatment of uncertainty. Most studies acknowledge that turnaround times are subject to high variability due to factors such as stochastic process durations, arrival delays, and resource constraints. It is important to note that this uncertainty is directly related to the prediction horizon. As shown in Table 3.1, most predictions are made at the outset or during the turnaround (tactical) process, when many operational factors are already known, thereby reducing uncertainty. Given this importance, Asadi et al. [13] and Wu & Caves [14] explicitly model uncertainties by using probabilistic distributions and stochastic optimisation techniques, respectively. This focus on uncertainty management reinforces the points made by [18], which highlights variability as a critical factor affecting operational efficiency.

The literature also indicates a growing trend towards integrating real-time data into turnaround predictions. Early studies relied primarily on historical data and static models, whereas recent approaches, such as those by Cui et al. [11] and De Falco et al. [15], adopt dynamic frameworks that continuously update predictions

Table 3.1: Summary of methods, horizons, inputs, uncertainties, future work, and key findings.

Citation	Methods	Horizon	Input Data	Uncertainties	Future Work	Findings
Cui et al. [11]	Time Transition Petri Net, Bayesian updates	Tactical (post on-block)	12 sub-process durations (e.g., deplaning, fueling), flight type	Stochastic durations via Bayesian probabilities	Include stand environment, resource constraints	RMSE: 3.75 min, MAE: 3.40 min (sub-process predictions)
Luo et al. [10]	ML classification (Decision Tree, RF) Fusion for sequential sub-processes	Tactical (post on-block)	Turnaround sub-process durations, airline, aircraft, arrival delay, time factors	Operational variability in durations and overlaps	Address data loss, update dynamic patterns	83.22% classification accuracy (RF)
Luo et al. [9]	Agent-based model (synthetic) ML (XGBoost, RF) with SHAP	Tactical (final TOBT update)	Synthetic/real sub-process durations, airline, aircraft, delays	Synthetic variability, feature uncertainty (SHAP)	Enhance synthetic realism, integrate real-time constraints	95% TOBT classification (RF), SD: 4.5–7 min
Asadi et al. [13]	Analytical convolution Chance-constrained MIP	Tactical (pre-departure)	Process distributions (Gamma/Weibull), TOBT	Durations (boarding, refueling), resource availability	Consider process dependencies, integrate real-time data	MIP achieves TOBT compliance (90%, 58–67 min)
Wu & Caves [14]	Analytical model with stochastic PDFs Cost optimization	Flight scheduling phase	Historical flight data, Beta-distributed arrivals, cost parameters	Arrival variability (Beta), delay propagation	Dynamic buffer adjustments, real-time adaptation	Optimal buffer varies by route
Fricke & Schultz [16]	Monte Carlo simulation Analytical modelling of critical paths	Tactical (during turnaround)	Ground ops data (3 German airports), process durations, delays	Inbound delays, resource limits	Dynamic buffer allocation, A-CDM integration	New buffering reduces delay propagation by ∼33%
AhmadBeygi et al. [17]	Linear programming Discrete-event simulation	Pre-tactical (flight scheduling)	Flight schedules, crew pairings, historical delay distributions	Root delays (mechanical/weather) & propagation via connections	Integrate crew/passenger links, model recovery decisions	Slack reallocation cuts delays by 25–50%
De Falco et al. [15]	XGBoost (probabilistic) Regression & classification	Strategic (days/hours pre-ops) Tactical	Scheduled times, arrival delay, congestion metrics	Probability distributions (95% intervals)	Generalise across airports, refine outlier detection	MAE: 3.8–5.9 min, RMSE: 5.0–8.4 min

as new data become available. This evolution is particularly important for tactical decision-making and operational adaptability, underscoring a move towards more dynamic models.

In summary, the review of the literature reveals an evolution from deterministic and static models towards dynamic and probabilistic approaches. The findings across these studies confirm the observations made in

[18] regarding the importance of real-time data integration, effective uncertainty management, and the use of advanced modelling techniques. Future research is encouraged to build on these insights, particularly by exploring the integration of diverse data sources and refining dynamic prediction models to further understand and act on operational variability.

3.2. DES & ABS

As shown in Section 3.1, a lot of work has been done around DES and ABS.

Firstly, the authors of [11] propose a Time Transition Petri Net (TTPN) model combined with Bayesian inference to improve the dynamic prediction of aircraft turnaround time. Their approach captures the sequential and parallel relationships between turnaround processes, using Bayesian probability to continuously update forecasts as new data becomes available. To validate their model, they analyse real operational data from a major hub airport in China, focusing on commonly used aircraft models such as the A320 and B777. Their method achieves a root-mean-square error (RMSE) of 3.75 minutes and a mean absolute error (MAE) of 3.40 minutes when compared to actual turnaround times, with MAE up to 6 minutes for predictions of individual processes. This approach is explicitly designed for real-time operational decision-making, continuously refining turnaround time predictions as processes unfold. The authors suggest that future research should focus on incorporating more real-world operational data to improve accuracy further, particularly addressing the challenges posed by data variability and the complex interdependencies within turnaround operations.

The authors of [10] propose a data-driven fusion model to predict aircraft turnaround time by integrating sequential dependencies among sub-processes (deplaning, unloading, fueling, etc.). Using real-world data from a European airport (22,080 valid 2019 records filtered to 20–60 minute turnarounds), they classify durations into four intervals (20–30, 30–40, 40–50, 50–60 minutes) via decision trees (76.32% accuracy) and random forests (83.22% accuracy). The model incrementally updates predictions by replacing historical sub-process duration estimates with real-time data as operations progress. A fusion logic categorizes sub-process overlaps as coverage, independence, or fusion (e.g., fueling often follows deplaning). A principal component analysis visualizes correlations between sub-process durations and domain knowledge (aircraft type, airline, arrival delay). Regression within intervals achieves RMSE values of 0.5–3 minutes, with higher errors in longer intervals (3 minutes RMSE for 50–60 minute turnarounds). Future work focuses on addressing incomplete sub-process data and expanding the model's applicability to diverse airports.

In [9], the same team develops an agent-based simulation to generate synthetic turnaround data (30,000 records) mimicking real-world sub-process interactions (e.g., parallel cleaning/catering and sequential baggage unloading/loading). They validate models against 22,620 actual 2019 records from a European airport. Using XGBoost, random forests, and decision trees, regression predictions achieve standard deviations as low as 4.5 minutes when incorporating sequential timestamps. For classification, random forests achieve 95% accuracy in predicting TOBT adherence within 10-minute intervals at the Airport Collaborative Decision Making (A-CDM) milestone preceding final off-block. SHAP analysis reveals that estimated stand occupancy and arrival delay are critical features, while sub-process durations have secondary impact. Synthetic data underperforms real data due to oversimplified agent rules (e.g., fixed sub-process sequences). Future work aims to refine agent-based simulations with stochastic resource constraints and integrate real-time weather data anomalies.

3.3. Uncertainty in Aircraft Turnaround Prediction

As discussed in Section 3.2, data pertaining to sub-processes is frequently incomplete or inaccurate, which renders precise predictions challenging. This is largely because the data is often derived from manual, handwritten inputs. Moreover, the turnaround process is influenced by a multitude of factors and is subject to various bottlenecks, potentially resulting in significant variability in turnaround times. This variability, coupled with the dependence on numerous factors, suggests that the quality of predictions is heavily contingent upon the prediction horizon. As highlighted in Section 3.1, most predictions are made at the outset or during the turnaround process, when many operational factors are already known, thereby reducing uncertainty. Consequently, it is important to address the impact of uncertainty on predictive accuracy.

The authors of [13] propose a mathematical optimization model that incorporates stochastic variations into aircraft turnaround prediction. They highlight that turnaround time is subject to random influences, such as passenger boarding behaviour, resource availability, and last-minute maintenance activities, all of which contribute to high variability in the estimated off-block time (EOBT). The study introduces two key methodologies: an analytical convolution approach, which aggregates stochastic process influences into a single probabilistic distribution, and a chance-constrained mixed-integer programming (MIP) model, which optimizes process execution to meet the Target Off-Block Time (TOBT) with a given confidence level (attaining a 90% compliance rate within 58–67 min). Their approach is tested on an Airbus A320 turnaround scenario, using process distributions derived from real-world operational data. The study is particularly relevant for day-of-operations decision-making, where real-time updates can help airport operators adjust scheduling dynamically to mitigate potential delays. The authors suggest that future research should focus on refining stochastic models by incorporating dependencies between sub-process durations, which were assumed independent in their approach.

The work of [14] further expands on the role of stochastic functions in aircraft turnaround optimization, particularly in the use of schedule buffer times to absorb operational uncertainties. They introduce a probabilistic approach where the arrival punctuality of inbound aircraft is modelled using stochastic probability density functions, capturing the uncertainty in flight schedules. Their study, validated with real flight data from a European carrier, demonstrates that proper allocation of schedule buffer time can significantly reduce departure delays while minimizing system costs. The research highlights the trade-off between aircraft utilization and delay minimization, showing that longer buffers improve punctuality but reduce overall productivity. Their findings emphasize that airlines must strategically adjust turnaround schedules based on route-specific arrival patterns, rather than applying a one-size-fits-all buffer.

3.4. Buffer and Slack Allocation in Aircraft Turnaround Planning

As previous research has shown, uncertainties in aircraft turnaround operations require systematic buffer allocation to minimize delay propagation. The role of buffer times in mitigating disruptions and maintaining schedule reliability has been widely discussed, particularly in the context of short-haul rotations, where delays accumulate over multiple legs. This section explores the research conducted on optimal buffer allocation strategies and their impact on turnaround performance.

The authors of [16] investigate the impact of delay propagation in turnaround operations and propose a dynamic buffer allocation strategy to improve schedule reliability. Using empirical data from Lufthansa and City Line at Frankfurt, Munich, and Leipzig airports, they analyse the critical turnaround path, identifying key sub-processes such as de-boarding, fuelling, catering, cleaning, and boarding as primary delay sources. Their findings reveal that current buffer strategies are heuristic and inconsistent, with turnaround times ranging from 30 to 55 minutes for A320-family aircraft and up to 2.5 hours for wide-body aircraft. To address this, they apply Monte Carlo-based stochastic modelling, demonstrating that optimized dynamic buffers can absorb up to 33% of delays while minimizing unnecessary ground time extensions by setting a better TOBT. Their model is designed for day-of-operations decision-making, particularly when operating under A-CDM frameworks. The study suggests that real-time buffer adjustments, based on inbound delay rather than static allocations, could reduce departure delays and improve resource efficiency. For future research, they recommend enhancing real-time monitoring systems and integrating machine learning techniques to dynamically adjust buffer sizes based on historical and real-time turnaround data.

A complementary approach is presented by [17], who focuses on reducing airline delay propagation by optimizing the allocation of scheduled slack. Their research highlights that secondary delays, caused by late-arriving aircraft, constitute over 33% of total flight delays in the U.S., which is partly caused by inefficient slack distribution across airline schedules. Using operational data from a major U.S. carrier, they propose a linear programming-based flight re-timing model, which redistributes existing slack to the most delay-sensitive connections without increasing overall schedule buffer time. Their approach optimizes departure times while maintaining feasibility constraints related to crew pairings, aircraft rotations, and passenger connections. Their results demonstrate that minor modifications in departure times, computed through simulation-based validation, significantly reduce downstream delay propagation, achieving up to a 50% reduction in propagated delays in some scenarios.

Both studies emphasize that buffer and slack allocation should not be static but dynamically optimized. Fricke et al. [16] propose real-time buffer adjustments within A-CDM frameworks, while [17] suggest reallocating slack during the scheduling phase to pre-emptively minimize disruptions. They propose that future research should explore machine learning techniques for adaptive buffer adjustments, integrating historical and real-time data to optimize slack and turnaround buffers dynamically.

3.5. Probability Distribution Models

Just as dynamically allocating slack can aid operational controllers and schedulers, a probabilistic prediction of turnaround time offers another valuable tool for decision-making. Beyond providing data-driven estimates of uncertainty, which directly address the challenges outlined in Section 3.3, such predictions serve a similar role to optimized slack allocation, offering controllers a clearer understanding of the likelihood that a flight will exceed its scheduled turnaround time, as will be shown in the following paragraph. This allows for proactive adjustments, helping identify the best-case minimum turnaround time while anticipating potential overruns, ultimately enhancing tactical scheduling and resource management.

The study by [15] introduces a probabilistic machine learning approach to predict aircraft turnaround time and TOBT, addressing key challenges in uncertainty estimation and operational predictability. Unlike deterministic models or DES, which struggle with data quality issues and real-world variability (as discussed in Section 3.3), this approach quantifies prediction uncertainty, offering more actionable insights for airport operators, ground handlers, and air traffic control (ATC). Using operational data from Prague, Geneva, Arlanda, and Fiumicino airports, the study demonstrates that probabilistic models can generalize across multiple locations while maintaining predictive accuracy. The authors develop a hybrid machine learning framework, combining XGBoost regression models with a classification-based probability distribution approach. This method extends predictions beyond single-point estimates, instead generating probability distributions of turnaround times, allowing for assessments of both best-case scenarios and potential overruns. Their models achieve mean absolute errors (MAE) between 8.4 and 7 minutes in the strategic/pre-tactical phase, improving to 6 minutes in the tactical phase when more accurate in-block data is available. A key advantage of this approach is its adaptability, enabling real-time adjustments within the A-CDM framework.

The study also highlights key variables influencing turnaround time predictions, identifying available turnaround time, aircraft type, and congestion levels as the most critical factors. Here, the congestion level represents the ratio of the hourly number of planned turnarounds on a specific day to their overall average values per airline. Additionally, a generalized model trained on data from all four airports performs comparably to airport-specific models, suggesting that a unified predictive approach could be effectively scaled to additional locations. The authors of [15] recommend expanding the model to more airports to assess its scalability and incorporating additional predictive features such as handling process durations, ground equipment availability, and weather conditions. They also emphasize the need for improved data-sharing between airlines, ground handlers, and airports to enhance model performance. By integrating these refinements, probabilistic machine learning could serve as a scalable, real-time decision-support tool, improving turnaround predictability and minimizing operational disruptions caused by last-minute TOBT changes.

4 Research Definition

This chapter establishes the foundation of the research by identifying the limitations in current approaches to aircraft turnaround time prediction and outlining the proposed contributions. Section 4.1 discusses the identified gaps in existing methodologies and motivates the need for a probabilistic machine learning framework that integrates real-time, multi-source data and dynamically quantifies operational uncertainty. Furthermore, the chapter presents the primary and sub-research questions in Section 4.2.

4.1. Research Gap and Proposed Contribution

Current approaches to aircraft turnaround time prediction face several limitations. While methodologies like DES and ABS [18] require granular sub-process data, which is often unavailable or unreliable in practice due to manual reporting, existing machine learning models predominantly rely on synthetic datasets [9] or deterministic outputs, failing to quantify operational uncertainty dynamically. Furthermore, stochastic methods [13] often assume sub-process independence, overlooking interdependencies in real-world operations, while buffer allocation strategies [16, 17] use static parameters and models, which are not well suited for dynamic day-of-operations decision-making. Lastly, probabilistic frameworks like [15] lack validation in heterogeneous, high-traffic environments and depend on fragmented data-sharing protocols. These gaps highlight the need for:

- **Real-time multi-source integration:** Unified models combining real-time ATC, weather, airline operational data, and more [9, 18]. Section 3.3 discussed that uncertainty is directly related to the prediction horizon, and Table 3.1 showed that most predictions are made during the tactical phase, when reliable operational data is available. Real-time data allows for to allow for dynamic predictions, before and during the tactical phase, whilst tracking the uncertainty.
- **Uncertainty-aware predictions:** Probability distributions, instead of single value estimates, to identify turnaround time distributions and quantify risks [13]. This increases the interpretability of the results, giving insights on both the minimum turnaround time, as well the expected turnaround time and the gap between those two. This allows to quantify the uncertainty in the prediction.
- **Operational validation:** Testing in live environments to assess prediction evolution as tactical phase comes closer [15]. Table 3.1 showed that the most common prediction horizon is during the turnaround itself, however, validation was mostly done on historical data rather then live environments. Furthermore, previous research rarely states a precise prediction horizon for their results, making it difficult to value their results for real-world operations.

This research addresses existing gaps by developing a probabilistic machine learning framework for aircraft turnaround time prediction, leveraging real-world data from a European hub airline. The model incorporates multi-source inputs, including ATC schedules, weather conditions, and aircraft configuration, while also employing dynamic updates to refine predictions as operations unfold, allowing stakeholders to proactively adjust buffers. The framework will be validated against operational data from an European airline, operating at a high-traffic hub airport, assessing prediction errors as the tactical phase approaches. By bridging the gap between theoretical models and practical implementation, this work aims to provide airlines with actionable, probabilistic insights into turnaround operations.

In contrast to the closest previous study by De Falco [15], discussed in Section 3.5, where predictions were made once at pre-tactical and once at tactical stages without explicit timestamps, this research brings the following advancements:

- **Enhanced Data Availability and Feature Set:** Whereas the approach in the work [15] was applied to data from four European airports with widely varying performance metrics, our model focuses on a single airline. This focus allows us to exploit a richer and more granular feature set that incorporates comprehensive operational data, thereby providing deeper insights into the determinants of turnaround times.
- **Dynamic Prediction Framework:** Although earlier models could be considered dynamic, offering a pre-tactical and tactical prediction, they did not specify the exact timestamps for prediction updates. This research advances this by developing a dynamic prediction framework that evaluates performance continuously at given timestamps leading up to the turnaround phase. As critical features, such as recent delay information, are updated, the prediction accuracy will increase and the associated uncertainty will decrease. This continuous updating process could equip operational controllers with a more reliable basis for real-time decision-making and slack allocation.
- **Airline-Specific Perspective:** Prior work adopted an airport-centric perspective, attempting to predict turnaround times across different airlines. This approach introduced significant variability, as operational practices differ markedly between low-cost carriers, non-hub airlines, and hub airlines. By concentrating on a single airline, this research's framework minimises such variability, enabling a more accurate and tailored analysis of turnaround operations.

4.2. Research Questions

Research Objective

The primary objective of this research is to develop and validate a machine learning framework for dynamic aircraft turnaround time distribution prediction that integrates real-time, multi-source data, and quantifies operational uncertainty.

Research Question 1

How accurately can a probabilistic machine learning framework dynamically predict aircraft turnaround time distributions?

To address the main research question (RQ 1), the study will explore the following sub-questions:

RQ1.1 What are the key operational and environmental factors affecting aircraft turnaround time?

RQ1.2 Which probabilistic modelling techniques are best suited to capture and quantify the inherent uncertainty in turnaround operations?

RQ1.3 What validation strategies and performance metrics can be employed to assess the framework's effectiveness using operational data?

RQ1.4 How does the predictive uncertainty evolve between the moment of take-off of the previous flight in the turnaround, and its landing?

5 Project Plan

This chapter presents an in-depth overview of the project plan. Section 5.1 details the overall methodological framework, covering the integration of diverse data sources (Section 5.1.1), preprocessing (Section 5.1.2), and feature engineering (Section 5.1.3). Furthermore, the chapter describes the use of predictive models (Section 5.1.4), including both tree-based and neural network approaches, and outlines the procedures for model training, hyperparameter tuning, and validation (Section 5.1.6). Finally, Section 5.2 summarises the project timeline and key milestones, guiding the sequential phases of the research.

5.1. Methodology

This section outlines the overall methodological framework for predicting aircraft turnaround time in a European hub airline context.

5.1.1. Data Source Overview

The airline under study offers multiple internal datasets, augmented by external sources (e.g. weather or airport traffic data). Table 5.1 provides an overview of the core datasets currently available or considered but ultimately not used, each containing attributes critical for modelling different factors affecting TAT.

Table 5.1: Overview of Available Datasets

Dataset	Description	Date Range
Flight leg Data	Contains operational details such as scheduled/actual arrival and departure times, flight numbers, gate assignments, delay codes, aircraft rotations	May 2021 – Present
Passenger Data	Includes forecasted and actual passenger loads	July 2021 – Present
Flight Plan	Provides cargo and baggage figures per flight	Aug. 2022– Present
Weather Data	Meteorological information, such as wind, temperature, and precipitation	Aug. 2023 – Present
Air Traffic Control (ATC) Data	Routing information, slot allocation, airspace constraints, estimated approach times	June 2022 – Present
Other Airline Data (AOG)	Information on airport movement of other airlines to measure airport congestions	Jan. 2024 - Present
ACARS Data (Never got access ultimately)	Aircraft communications addressing and reporting system timestamps (e.g. door closure)	Not confirmed

It is important to note that most of these datasets are both historised, and available real-time. This allows for training and testing in real day of operations scenarios, as will be discussed further in Section 5.1.6.

5.1.2. Data Preprocessing

Before model development, several preprocessing steps are undertaken to ensure data consistency and reliability. These tasks include:

- **Data Cleaning:** Removal of duplicate records and correction of outliers (e.g., obviously incorrect timestamps).
- **Data Joining:** Joining of disparate data sources to construct a coherent operational timeline.
- **Data Sampling:** Filtering the dataset to focus on flights with *unconstrained* turnaround times. Specifically, flights subject to allocated slots, overnight ground times, or delays waiting for connecting passengers are excluded, so that the turnaround duration itself is the primary driver of operational performance. This ensures that model training and validation remain centred on the cases where turnaround processes directly influence airline operations.

This systematic preprocessing, including targeted data sampling, preserves the full variability of relevant real-world operations and ensures the model is trained on cases where turnaround efficiency truly matters.

5.1.3. Feature Engineering

Feature engineering plays a dual role in this framework by both constructing new predictors and evaluating their relevance.

Firstly, new features are derived to capture key operational nuances such as airport congestion level and binary weather condition flags.

Secondly, understanding the contribution of individual predictors is crucial for both refining the predictive model and ensuring its interpretability. In this study, a multi-pronged approach is adopted to evaluate feature importance:

- **Correlation Matrix Analysis:** A correlation matrix is constructed to examine pairwise linear relationships between features and the target variable. This exploratory analysis assists in identifying strongly correlated predictors and detecting potential multicollinearity, which can adversely affect model stability.
- **Permutation Importance:** This method involves randomising the values of each feature and measuring the consequent increase in the model's prediction error. The resulting metric provides a straightforward assessment of a feature's impact on the overall model performance [19].
- **SHAP Values:** The SHapley Additive exPlanations (SHAP) framework is employed to assign a game-theoretic value to each feature's contribution, both at the global and local levels. By aggregating SHAP values across the dataset, the analysis offers a detailed insight into how individual features drive model predictions [20].

Integrating these complementary methods not only aids in optimising the feature set by highlighting the most influential predictors but also enhances the transparency and interpretability of the predictive framework.

5.1.4. Model Development

To address the gaps identified in the literature, a probabilistic modelling framework is proposed that aims to forecast full distributions of turnaround time outcomes rather than providing single point estimates. This approach recognises the inherent uncertainty in operational data. Two broad classes of models are explored: tree-based ensemble methods and neural network-based approaches. Each family offers distinct benefits in capturing complex data relationships and quantifying uncertainty.

Tree-Based Ensemble Methods

Tree-based ensemble methods combine multiple decision trees to improve generalisation on structured (tabular) data. The idea is to partition the feature space into regions with decision-tree splits and then aggregate the outputs of many trees to approximate the target function. Formally, for a feature vector $\mathbf{x} \in \mathbb{R}^d$, an unweighted ensemble of M trees predicts according to Equation 5.1, whereas a boosting ensemble uses the weighted form shown in Equation 5.2. In both cases $T_m(\mathbf{x})$ denotes the output of the m-th tree and the ensemble parameters (tree structures, split points, and, where applicable, weights β_m) are estimated from training data.

$$\hat{y} = f(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} T_m(\mathbf{x}), \tag{5.1}$$

$$\hat{y} = \sum_{m=1}^{M} \beta_m T_m(\mathbf{x}), \tag{5.2}$$

Random Forests (Bagging): Random Forests [19] are a canonical bagging approach. Each tree is grown on a bootstrap sample of the training set and considers a random subset of predictors at every split, which decorrelates individual trees. The resulting forest, typically comprising hundreds of deep trees, is a high-variance collection whose averaged prediction exhibits markedly lower variance than any single tree. Random Forests perform competitively "out of the box", require little hyper-parameter tuning, cope well with nonlinearities and interactions, and provide heuristic measures of feature importance.

Gradient-Boosted Decision Trees: Boosting builds trees sequentially, each new tree correcting the residual errors of the current ensemble. Gradient-boosted decision trees (GBDT) [21] implement this by fitting successive trees to the negative gradient of a chosen loss function. Shrinkage (a learning rate), limits on tree depth, and penalties on leaf weights act as regularisation to curb overfitting. Modern implementations dominate tabular benchmarks: *XGBoost* [22] employs second-order optimisation, L_1/L_2 regularisation and sparse-aware split finding, while *LightGBM* [23] accelerates training via histogram-based splits, leaf-wise growth, Gradient-Based One-Side Sampling and Exclusive Feature Bundling. Both libraries offer early stopping, monotonic constraints and native quantile objectives. In practice, boosting ensembles comprise hundreds to thousands of shallow trees, weighted by the learning rate, and primarily reduce bias; their performance is sensitive to careful hyper-parameter tuning.

Probabilistic Predictions: Standard tree ensembles output point estimates, yet many forecasting tasks require predictive distributions. *Quantile Regression Forests* [24] replace the leaf average by conditional quantiles, yielding prediction intervals directly from a Random Forest. Analogously, GBDT models can optimise a pinball (quantile) or other distributional loss, either for individual quantiles or jointly for multiple quantiles, to approximate the full conditional distribution. These adaptations furnish uncertainty estimates alongside point forecasts, thus rendering tree-based ensembles suitable for probabilistic prediction.

Neural Network-Based Approaches

> Note: Neural network-based approaches were ultimately deprioritised in the thesis work due to the lack of proven model performance improvement compared to tree-based methods in existing work, as well as the reduced explainability of such methods.

Neural networks offer a flexible framework for approximating complex, non-linear relationships within the data. A popular strategy is to configure a network to output the parameters of a probability distribution, often achieved via a mixture density network (MDN) [25]. However, the neural network approach is not restricted to MDNs; alternative architectures, such as Bayesian neural networks [26] or models leveraging variational inference [27], may also be utilised to model the complete probability distribution of TAT. In general, the conditional distribution can be formulated as:

$$p(T \mid \mathbf{x}) = \sum_{k=1}^{K} \pi_k(\mathbf{x})\, \mathcal{D}_k\big(T \mid \theta_k(\mathbf{x})\big),$$

where π_k denotes the mixing coefficients and $\mathcal{D}_k$ represents a selected probability distribution (e.g. Gaussian) parameterised by θ_k. This formulation enables the capture of uncertainty and potential multimodality in the TAT outcomes.

Final model selection will be guided by the performance metrics outlined in Section 5.1.6, with the best-performing model, or an ensemble thereof, recommended for operational use.

5.1.5. Model Training & Hyperparameter Tuning

Model training and hyperparameter tuning are critical for optimising predictive performance. The training procedure involves:

- **Training:** Optimising model parameters using historical operational data samples, which are selected as discussed in Section 5.1.2.
- **Hyperparameter Tuning:** Adjusting key parameters (e.g., tree depth, regularisation terms, learning rates) to enhance model generalisation.
- **Model Selection:** Comparing the performance of tree-based ensembles and neural network architectures using established performance metrics.

This iterative process ensures that the chosen model, whether standalone or an ensemble, achieves both high accuracy and reliable TAT distributions.

5.1.6. Validation & Testing

Robust validation is essential for demonstrating both scientific and operational viability of the proposed TAT prediction framework. Two complementary approaches are proposed: *traditional validation* using standard train-validation-test splits and *operational validation* designed to reflect real-world usage scenarios.

Traditional Validation

The standard practice in machine learning research divides the data into distinct training, validation, and test subsets. This approach provides an unbiased estimate of model performance on unseen data. The workflow is:

- **Training:** Optimise model parameters on historical records.
- **Validation:** Tune hyperparameters (e.g. tree depth, regularisation terms, learning rate).
- **Testing:** Report final performance metrics (e.g. mean absolute error, prediction interval coverage) on a hold-out dataset.

While this procedure ensures a scientifically sound comparison of models, it may not accurately capture the real-world environment due to data drift and biases in how inputs arrive over time. For example, in airline operations, there might be more data points recorded by the flight management system closer to arrival or departure, thus artificially inflating the apparent accuracy, since these model tend to be more accurate close to arrival or departure, as confirmed by the work of [15].

Operational Validation

To complement the standard approach, an operational validation strategy is implemented. This strategy simulates how predictions would be generated and used in real-time:

- **Timed Predictions:** The prediction for a certain flight will be stored at two time points, as defined in Section 2.3. This replicates the real operational scenario where turnaround estimates are repeatedly updated as new information (e.g. updated ETAs, changes in load) becomes available.
- **Performance Tracking:** Each prediction is compared against the eventual ground-truth TAT. Errors are tracked over time, illustrating how the model's confidence interval narrows (or not) as departure approaches.
- **Bias Detection:** By aligning predictive performance with operational timestamps, biases introduced by frequent late-flight data entries can be detected and mitigated.

This two-tier approach ensures that the model is both scientifically sound and practically deployable. In particular, operational validation helps identify cases where a model may appear accurate on a retrospective dataset but underperforms when subjected to on-the-fly, real-world usage patterns.

5.1.7. Results & Analysis

Benchmarking against existing research is challenging for two primary reasons. First, studies employ different datasets, leading to variations in outcomes. Second, and more critically, previous research like [15], does not specify the precise time point at which predictions were made. Given that uncertainty diminishes as operations unfold, the timing of predictions significantly influences the results, making fair comparisons difficult.

To address these issues, this thesis adopts a different strategy by benchmarking against a model developed by the airline under investigation. This model used similar data and was evaluated at the same prediction horizon, thereby offering a more equitable basis for comparison. Table 5.2 presents the baseline results from the airline's model:

Table 5.2: Baseline Performance Metrics from the Airline Model

Timepoint	MAE [min]	RMSE [min]
Previous Flight Take-Off	17.5	41
Previous Flight Landing	12	34

These values serve as the baseline that the current research aims to meet or exceed. As detailed in Section 2.3, the evaluation focuses on these two prediction horizons.

The evaluation of the proposed approach will centre on two key aspects:

- **Prediction Accuracy:** The proximity of the predicted mean TAT to the actual TAT will be assessed at various time points preceding departure, using standard error metrics such as Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).
- **Uncertainty Quantification:** It is anticipated that the uncertainty in the predictions will decrease as the turnaround comes closer. This will be evaluated by examining the predicted variance.

Furthermore, although it remains uncertain which model architecture will ultimately demonstrate superior performance, a parallel evaluation of tree-based and neural network-based approaches is expected to reveal complementary strengths. As departure time nears and key features are updated, improvements are anticipated in both the accuracy of the predicted mean and the confidence in the uncertainty estimates. These enhancements are intended to provide operational controllers with a robust, real-time basis for decision-making and slack allocation.

The forthcoming results will be presented through a comprehensive set of graphs and statistical measures that reflect these evaluation criteria, thereby offering detailed insights into the predictive capabilities of the proposed framework.

5.2. Planning

The research project is organized into four sequential phases, as illustrated in the Gantt chart (Figure 5.1). Key deliverables and checkpoints are distributed throughout the timeline to ensure systematic progress monitoring.

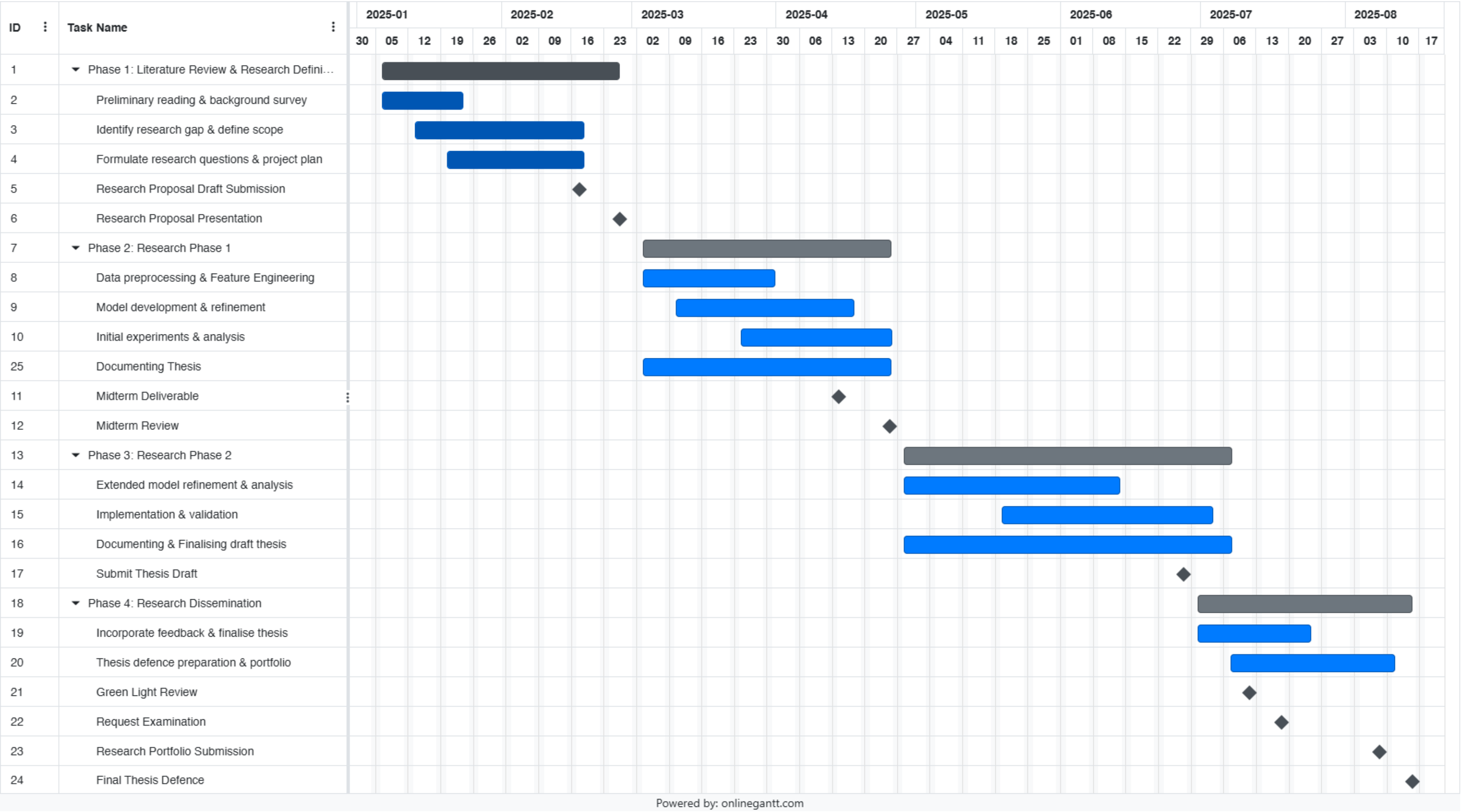
Key Milestones

- **Research Proposal Draft Submission**: 17 February 2025
- **Research Proposal Presentation**: 26 February 2025
- **Midterm Deliverable**: 14 April 2025
- **Midterm Review**: 25 April 2025
- **Thesis Draft Submission**: 27 June 2025
- **Green Light Review**: 11 July 2025
- **Research Portfolio Submission**: 8 August 2025
- **Final Thesis Defence**: August/ September 2025

Project Phase Overview

1. **Literature Review & Research Definition** (January-February 2025):
   Establishing theoretical foundations through comprehensive background study, gap identification, and formulation of research questions.
2. **Research Phase 1** (March-April 2025):
   Initial implementation stage featuring data preparation, model development, and preliminary experimentation. Includes midterm progress evaluations.

3. **Research Phase 2** (April-July 2025):
   Extended model refinement, validation processes, and completion of thesis documentation. Culminates in draft thesis submission.
4. **Research Dissemination** (July-August 2025):
   Final revisions incorporating feedback, examination preparation, and formal research outcomes presentation through defense.



ID	Task Name
1	Phase 1: Literature Review & Research Defini…
2	Preliminary reading & background survey
3	Identify research gap & define scope
4	Formulate research questions & project plan
5	Research Proposal Draft Submission
6	Research Proposal Presentation
7	Phase 2: Research Phase 1
8	Data preprocessing & Feature Engineering
9	Model development & refinement
10	Initial experiments & analysis
25	Documenting Thesis
11	Midterm Deliverable
12	Midterm Review
13	Phase 3: Research Phase 2
14	Extended model refinement & analysis
15	Implementation & validation
16	Documenting & Finalising draft thesis
17	Submit Thesis Draft
18	Phase 4: Research Dissemination
19	Incorporate feedback & finalise thesis
20	Thesis defence preparation & portfolio
21	Green Light Review
22	Request Examination
23	Research Portfolio Submission
24	Final Thesis Defence

Figure 5.1: Gantt Chart for Project Timeline

References

[1] *Switchfly Identifies the Financial Damage of Irregular Operations for Airlines*. Accessed: 2025-01-27. 2017. URL: https://www.prnewswire.com/news-releases/switchfly-identifies-the-financial-damage-of-irregular-operations-for-airlines-300561616.html.

[2] *Managing Irregular Operations at Airports: Problems and Solutions*. Accessed: 2025-01-27. 2023. URL: https://blog.adbsafegate.com/managing-irregular-operations-at-airports-problems-and-solutions/.

[3] *The True Cost of Aviation Disruption: A Closer Look*. Accessed: 2025-01-27. 2024. URL: https://www.cmacgroup.com/blog/the-true-cost-of-aviation-disruption-a-closer-look.

[4] *Innovative Airline Operations: The Turnaround*. Accessed: 2025-01-27. 2023. URL: https://www.oag.com/blog/innovative-airline-operations-the-turnaround.

[5] Eindhoven Airport. *Eindhoven Airport Uses AI to Improve Turnaround Process*. Accessed: 21-Feb-2025. 2023. URL: https://www.aviationpros.com/ground-handling/press-release/53077267/eindhoven-airport-eindhoven-airport-uses-ai-to-improve-turnaround-process.

[6] Passenger Terminal Today. *How is AI revolutionizing airports around the world?* Accessed: 21-Feb-2025. 2024. URL: https://www.passengerterminaltoday.com/features/exclusive-feature-how-is-ai-revolutionizing-airports-around-the-world.html.

[7] EUROCONTROL. *EUROCONTROL Data Snapshot: Reactionary delays and their impact on airline operations*. Tech. rep. Accessed: 10-February-2025. EUROCONTROL, 2023. URL: https://www.eurocontrol.int/publication/reactionary-delays-data-snapshot.

[8] Iain Coutts et al. *Turnaround: Here's What Happens When An Aircraft Is On The Ground*. Accessed: 2025-02-22. 2024. URL: https://simpleflying.com/aircraft-turnaround-process/.

[9] Mingchuan Luo et al. "Agent-based Simulation for Aircraft Stand Operations to Predict Ground Time Using Machine Learning". In: *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)* (2021), pp. 1–10. DOI: 10.1109/DASC52595.2021.9594325.

[10] Mingchuan Luo et al. "Data-driven Fusion of Turnaround Sub-processes to Predict Aircraft Ground Time". In: *Air Transport Research Society 2022 (ATRS)* (2022). URL: https://www.researchgate.net/publication/363153261_Data-driven_fusion_of_turnaround_sub-processes_to_predict_aircraft_ground_time.

[11] Yanyu Cui et al. "Aircraft Turnaround Time Dynamic Prediction Based on Time Transition Petri Net". In: *PLOS ONE* 19.7 (2024), e0305237. DOI: 10.1371/journal.pone.0305237.

[12] Y. Li et al. "Dealing with Uncertainty: A Survey of Theories and Practices". In: *IEEE Transactions on Knowledge and Data Engineering* 25.11 (2012), pp. 2463–2482. DOI: 10.1109/TKDE.2012.105.

[13] Ehsan Asadi et al. "Coping with Uncertainties in Predicting the Aircraft Turnaround Time at Airports". In: *Operations Research Proceedings 2019* (2020), pp. 773–780. DOI: 10.1007/978-3-030-48439-2_94.

[14] Cheng-Lung Wu et al. "Modelling and Optimization of Aircraft Turnaround Time at an Airport". In: *Transportation Planning and Technology* 27.1 (2004), pp. 47–66. DOI: 10.1080/0308106042000184454.

[15] Paolino De Falco et al. "Probabilistic Prediction of Aircraft Turnaround Time and Target Off-Block Time". In: *Proceedings of the 13th SESAR Innovation Days* (2023). URL: https://www.sesarju.eu/sites/default/files/documents/sid/2023/Papers/SIDs_2023_paper_26%20final.pdf.

[16] Hartmut Fricke et al. "Delay Impacts onto Turnaround Performance: Optimal Time Buffering for Minimizing Delay Propagation". In: *Proceedings of the 8th USA/Europe Air Traffic Management

Research and Development Seminar (ATM 2009) (2009). URL: https://www.researchgate.net/publication/262567633_Delay_Impacts_onto_Turnaround_Performance_-_Optimal_Time_Buffering_for_Minimizing_Delay_Propagation.

[17] Shervin AhmadBeygi et al. “Decreasing Airline Delay Propagation by Re-Allocating Scheduled Slack”. In: *IIE Transactions* 42.7 (2010), pp. 478–489. DOI: 10.1080/07408170903468605.

[18] Michael Schmidt. “A Review of Aircraft Turnaround Operations and Simulations”. In: *Journal of Air Transport Management* 63 (2017), pp. 34–40. DOI: 10.1016/j.jairtraman.2017.05.003.

[19] Leo Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32. DOI: 10.1023/A:1010933404324.

[20] Scott M. Lundberg et al. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems* 30 (2017). DOI: 10.48550/arXiv.1705.07874.

[21] Jerome H Friedman. “Greedy function approximation: a gradient boosting machine”. In: *Annals of Statistics* 29.5 (2001), pp. 1189–1232. DOI: 10.1214/aos/1013203451.

[22] Tianqi Chen et al. “XGBoost: A Scalable Tree Boosting System”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), pp. 785–794. DOI: 10.1145/2939672.2939785.

[23] Guolin Ke et al. “LightGBM: A Highly Efficient Gradient Boosting Decision Tree”. In: ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 3146–3154. URL: https://www.researchgate.net/publication/378480234_LightGBM_A_Highly_Efficient_Gradient_Boosting_Decision_Tree.

[24] Nicolai Meinshausen. “Quantile Regression Forests”. In: *Journal of Machine Learning Research* 7 (2006), pp. 983–999. URL: https://dl.acm.org/doi/10.5555/1248547.1248582.

[25] Christopher M. Bishop. *Mixture Density Networks*. Tech. rep. Neural Computing Research Group, Aston University, 1994. URL: https://publications.aston.ac.uk/id/eprint/373/1/NCRG_94_004.pdf.

[26] Daniel T. Chang. “Probabilistic Deep Learning with Probabilistic Neural Networks and Deep Probabilistic Models”. In: *arXiv preprint* (2021). DOI: 10.48550/arXiv.2106.00120.

[27] PyMC Developers. *Variational Inference: Bayesian Neural Networks*. Accessed: 2025-02-14. 2024. URL: https://www.pymc.io/projects/examples/en/latest/variational_inference/bayesian_neural_network_advi.html.

Note to Reader

The remainder of this thesis comprises three parts, each addressing a different objective of the research.

Part II – Scientific Article presents the main results of the study in the form of a self-contained paper. This article focuses on the scientific contribution of the work, providing a comprehensive account of the model development, its predictive performance, and the implications for turnaround forecasting.

Part III – Operational Validation complements the article by examining the model's practical application in an operational setting. It addresses one of the original research questions by investigating how well the model performs when deployed in real-time, and identifies failure modes and limitations that may arise in practice.

Part IV – Closure offers concluding reflections. It returns to the overarching research questions, evaluates the main findings, and outlines recommendations for future work.

Together, these parts aim to convey both the scientific merit and the operational relevance of the developed approach.

Part II

Scientific Article

Dynamically Predicting Airliner’s Turnaround Time

Camille Derie

Delft University of Technology, Delft, The Netherlands

Abstract

Accurate prediction of aircraft turnaround time (TAT) is essential for mitigating reactionary delay, yet present methods remain constrained. Existent work uses discrete event simulations to predict individual ground activities but accumulate error and uncertainty, and in turn, other data-driven studies still provide a single static point estimate that is not updated with the latest operational information. It is therefore difficult to validate the use of these models in real operational environments. Although rolling, continuously updated forecasts have recently been explored for departure delay prediction, no study has yet extended such dynamic modelling to the turnaround itself, leaving a critical gap in operational decision support, as departure delay may include additional delay sources external to the turnaround process. This study develops and operationally tests a probabilistic machine learning framework that continuously updates full predictive distributions of TAT for a European hub carrier. Real time airline, meteorological and air-traffic feeds are merged into gradient boosting tree ensembles trained with quantile regression. Evaluation on past turnarounds yields a median absolute error of 10 minutes immediately after the preceding take-off, falling to 8 minutes at on-block. Results show that the uncertainty of the prediction reduces by a quarter as updated operational data like delays or air-traffic control slots come in. These findings show that uncertainty in TAT can be quantified accurately and in near real-time using data streams already present in airline operations, enabling controllers and optimisation engines to target mitigation measures proportionately and thereby reducing cascading delay, cost, and emissions.

1. Introduction

Irregular airline operations impose sizeable financial, operational, and environmental burdens on carriers worldwide. According to Eurocontrol, schedule disruptions due to air traffic congestion, weather, and staffing shortages are projected to have cost European airlines approximately €2.8 billion in 2024 alone [1]. Disruptions also increase fuel burn and carbon dioxide output; a universal reduction of auxiliary-power-unit use by only one minute per flight could remove more than 172 million kg of CO_2 annually across the commercial fleet [2]. A crucial factor that influences this financial loss and the excess emissions is aircraft turnaround time (TAT), the interval between on-block and off-block.

TAT is inherently variable because it is governed by a tightly choreographed, yet weakly synchronised set of ground activities. Even modest overruns can trigger rotational reactionary delays, which account for nearly 90% of all reactionary delay minutes at major European hubs [3]. Large network carriers are especially exposed: their banked wave structure concentrates demand for stands, equipment, and personnel, magnifying the effect of every minute of ground delay. Airports and airlines have been working on monitoring and predictive solutions, to enhance operations decision making and manage disruptions effectively [4, 5].

Academic work has approached TAT prediction from different directions. Firstly, discrete-event and agent-based simulations have tried modelling each sub-process in detail, yielding insight into bottlenecks but accumulating error across step simulations [6, 7]. Secondly, stochastic optimisation frameworks address uncertainty explicitly yet often rely on strong assumptions on independence between sub-processes [8]. More recently, data-driven models have emerged. For example, probabilistic tree ensembles can provide full predictive distributions, as demonstrated for several European airports [9]. However, those studies offer only one or two prediction updates on the day of operation, and do not provide a clear horizon of how far in advance of departure the prediction was made. Furthermore, they depend on heterogeneous multi-airport data sets, and supply limited evidence of performance in live settings.

A research gap therefore remains for a TAT-prediction framework that is both dynamic and probabilistic, provides clear prediction horizons starting hours before the departure of the previous flight, that assimilates rich real-time airline data, and that is validated within the operational environment of a hub carrier. Addressing this gap would provide continuously updated probability distributions that expose not only the expected turnaround duration but also the window of certainty. This information could both be useful for an operations controller, or as input to an operations optimization engine. The main contribution of this paper is thus the development and operational validation of a probabilistic supervised machine-learning framework that delivers continuously updated distributions of aircraft turnaround time for a European hub airline, integrating multi-source real-time data, including flight operations, weather, air-traffic control (ATC), and other airline's schedule data, whilst quantifying predictive uncertainty. Lastly, tree based ensemble methods are used thanks to their superior explainability, an important attribute for practical operational deployment.

This paper is organised as follows. Section 2 reviews prior work on turnaround-time prediction and clarifies the research gap. Section 3 details the data architecture, feature engineering, learning algorithms and uncertainty-quantification scheme. Section 4 explains the chronological train–test split and live-like validation platform, while Section 5 sets out the hypotheses that steer the evaluation. The results, including accuracy, calibration and feature-importance analyses, are presented in Section 6, after which Section 7 traces weekly performance and walks through two illustrative use cases. Section 8 reflects on the findings, limitations and future research directions, and Section 9 summarises the contributions and operational implications.

2. Literature Research

Efficient aircraft turnaround, the time between a flight's on-block and off-block, is widely recognised as a decisive lever for network punctuality, cost control, and environmental performance[10]. Variability in the duration of ground operations remains a principal cause of rotational reactionary delay, especially for hub-and-spoke carriers that operate tightly banked schedules. This section discusses past work done in Subsection 2.1, and formulates the research gap in Subsection 2.2.

2.1 Past Work

Early research relied on deterministic analytical formulations and on discrete event or agent based simulations. These approaches decomposed the turnaround into individual activities such as deboarding, refuelling, and catering, then propagated their estimated durations through critical path logic in order to predict the off block moment [11, 7, 6]. Recent advancements to this activity based perspective have been achieved by Tang *et al.*, who applied a cascaded gradient boosting framework to refresh milestone predictions continuously throughout the turnaround itself [12]. Although such models provided valuable insight into bottlenecks, they required granular timestamp data, often captured manually, and the separate error terms attached to every activity accumulated across the many sub processes that characterise a hub operation [10].

To address the evident variability, later studies embedded stochastic descriptions directly within either analytical convolution or chance constrained optimisation frameworks. Work on probabilistic buffer sizing and slack redistribution demonstrated that judicious time allowances can curb the reactionary effects of primary delay [13, 14, 8]. Nevertheless, these contributions treat the probability turnaround time distribution as an external input rather than as a model output and remain essentially static; once the slack is fixed, the method does not adapt to unfolding operational information.

The most recent shift has been towards probability distribution prediction through machine learning. De Falco *et al.* showed that tree based ensembles can deliver full turnaround distributions and narrower prediction intervals than deterministic or simulation techniques [9]. However, their study produced only two forecasts per turnaround: one in the pre-tactical window and one after the aircraft reached the gate. This split was based solely on the availability of a single feature, namely the updated on-block time of the previous flight. As a result, many relevant operational updates were ignored, leaving the forecasts largely static. Moreover, the authors combined data from airports with different operating practices, which limited airline-specific insights and masked how predictive uncertainty evolves over time. Finally, none of the existing work has been tested in an operational environment. As a result, they only demonstrate theoretically achievable performance, while in a live environment, numerous factors such as data leakage or drift can influence actual performance. This highlights the gap between theoretical solutions and practical applications.

Parallel work on departure delay has reinforced the value of probabilistic, data-driven prediction. Dalmau *et al.* [15] enhanced take-off-time estimates with explainable gradient-boosted decision trees, trimming the mean absolute error by roughly 2 minutes one hour before off-block while revealing the influence of ATM slotting, available turnaround time, and meteorological restrictions. Mori [16] shifted the focus from point estimates to full off-block distributions by fitting a Johnson-SU curve whose parameters were updated continuously through a shallow network that digests successive Target Off-Block-Time revisions. Their work showed potential to improve airport departure management, reducing average taxi-out queuing at a regional French hub by 8 %. Beltman *et al.* [17] extended the rolling-horizon principle by generating full delay distributions at six horizons from 90 to 15 minutes before scheduled departure. Their random-forest ensemble, trained on three years of hub-carrier operations, achieved a mean absolute error of 8.5 minutes at the 90-minute horizon and contained the realised delay for over three-quarters of flights. Together these studies confirm that decision-tree and neural-network ensembles can fuse heterogeneous, real-time data feeds into probabilistic forecasts, yet they remain departure-centric: early arrivals or buffered ground times collapse to zero delay and therefore vanish from the learning target.

By explicitly predicting the entire turnaround interval rather than the departure delay alone, the present work preserves this methodological lineage while probing further into the gate phase. This perspective captures the mitigating effect of scheduled slack when an inbound flight lands behind time and, conversely, exposes the risk that an on-time arrival still fails to depart punctually because ground-handling overruns. It therefore complements the delay-focused literature by linking probability forecasts to the controllable ground processes that ultimately determine network resilience.

2.2 Research Gap

The methodology proposed in this paper advances the current turnaround prediction work in three respects. First, it fuses multi source and real-time information, including airline flight leg and passenger records, weather observations, and air traffic control constraints. Secondly, it creates a probabilistic learning framework that updates continuously from hours before operations until departure, thus providing a clear prediction horizon. Third, the model was deployed in a live operational setting within the airline data pipeline, enabling continuous validation that its predictive performance can be achieved in practice rather than remaining solely theoretical. The resulting framework is designed not only to refine academic understanding of turnaround dynamics, but also to furnish airline controllers with actionable, uncertainty-aware predictions that support proactive decision-making and the mitigation of cascading delays. Moreover, by attaching confidence intervals to each forecast, the same predictions can be fed directly into airline optimisation engines, allowing them to weigh alternative recovery actions in proportion to the quantified risk.

3. Methodology

The objective of this study is to develop a dynamic, uncertainty-aware framework for predicting aircraft TAT across an extended prediction horizon, starting several hours before the turnaround and continuing throughout its progression. The model simultaneously quantifies the associated uncertainty. The data landscape is introduced in Subsection 3.1, where the structure, provenance, and preprocessing of the raw streams are described in Subsubsection 3.1.1. Subsequent paragraphs discuss feature engineering, and the procedure for feature selection and importance assessment (Subsection 3.2). The distinct data sources feeding the framework are detailed in Subsubsection 3.1.2, and the most influential predictive variables are summarised in Subsubsection 3.1.3. The modelling strategy is explained in Subsection 3.3, followed by the scheme adopted to quantify predictive uncertainty (Subsection 3.4). The tuning of hyperparameters is outlined in Subsection 3.5, whereas the prediction horizon and evaluation points are defined in Subsection 3.6. Finally, the performance metrics used to judge the models are set out in Subsection 3.7.

3.1 Data Landscape

This subsection outlines the structure, provenance, and transformation of the data that support the predictive framework.

3.1.1 Data Structure and Preprocessing

The predictive framework relies on a real-time, multi-source dataset that consists of about 50 operational updates on average throughout each aircraft rotation, combing data from the previous and departing flight. Each update is timestamped with the moment it becomes available, ensuring that every prediction reflects only information that would have been accessible at that precise point in time; this discipline is fundamental in avoiding data leakage. The data covers turnarounds of passenger flights at an international European hub.

Preprocessing begins with temporal alignment and integration of the disparate data sources, discussed in Subsubsection 3.1.2. Maintaining the chronological order of updates preserves the real-time flow of information on which the model later depends. Following integration, exploratory analysis examines distributions, reveals anomalies, and identifies missing values or inconsistencies. Insights gained guide data quality assurance and inform the construction of informative derived variables.

The dataset is then restructured so that each row represents an aircraft rotation rather than an individual flight leg. Information from the inbound leg, the outbound leg, and the intervening period is consolidated, producing a single record that captures the turnaround of interest. Overnight rotations, where the elapsed interval spans calendar days, are excluded because their operational constraints differ markedly from standard turnarounds.

Finally, time-aligned indicators map each database update to its relative position within the turnaround timeline. These engineered variables capture the temporal dynamics of the process and enable the model to adapt predictions to the progressive availability of information.

3.1.2 Data Sources

Four principal data streams inform the turnaround prediction model, each providing distinct operational or environmental context.

Flight Leg Data: Scheduled and actual gate times, routing information, passenger numbers, and cargo volumes describe the demand on ground services and expose discrepancies between planned and realised operations. Derived variables such as available turnaround time and aircraft rotation sequence shape the feasible pace of ground activity. Furthermore, information like airport terminal, gate, and aircraft type provide additional information to classify the type of turnaround.

Weather Data: Observations and forecasts of wind, visibility, and precipitation characterise both apron conditions and the wider air-traffic environment. These signals alert the model to circumstances in which ground teams must slow for safety, de-icing may be required, or runway capacity is reduced, all of which tend to prolong stand occupation.

ATC Data: Air-traffic control messages, notably slot assignments and delay notifications, reflect constraints in the surrounding airspace and at the runway. Features such as expected arrival delay and the Calculated Take-Off Time (CTOT) indicate the effective ground time available and the likelihood of holding at the gate before clearance.

Other Airline Data: Schedules and status updates for concurrent flights operated by other carriers act as a proxy for airport congestion. Metrics capturing movement counts and their average delays quantify competition for shared resources, enabling the model to adjust its estimate when the apron is busy.

3.1.3 Key Predictive Features

Table 1 lists the subset of features that emerge most prominently in the importance analysis. A complete inventory of features appears in the Appendix in Table 4. It should be noted that the complete feature set proved to be essential for capturing rare operational edge cases that, although influential in specific scenarios, do not appear prominently in subsequent feature–importance assessments.

Table 1. Most influential predictive features

Feature	Dynamic	Earliest information available
Available ground time	Yes	Before previous take off
Actual Off block delay previous flight	Yes	After previous take off
Expected departure delay of previous flight	Yes	Before previous take off
Expected off block delay of next flight	Yes	After previous take off
CTOT minutes after departure next flight	Yes	Before previous take off
Passengers next flight	Yes	Before previous take off
Previous departure airport	No	Before previous take off
Current destination airport	No	Before previous take off
Widebody aircraft indicator	No	Before previous take off

3.2 Feature Engineering, Selection and Importance Assessment

The modelling dataset incorporates a wide range of operational and environmental attributes, including scheduled and actual times, aircraft characteristics, and meteorological observations. To enrich the input space, several additional predictive variables are engineered. Indicators of airport congestion quantify inbound and outbound activity within rolling windows, while binary weather flags denote significant precipitation, strong winds, or reduced visibility. Furthermore, temporal relationships are represented through time-to-event metrics that measure the elapsed interval since key milestones, such as the previous flight's landing or take-off. Further variables reflect airline-specific practices, drawing on domain knowledge of scheduling patterns, delay propagation, and service routines. In addition to numerical predictors, categorical attributes such as aircraft subtype, airline, stand position, and recorded delay reason are incorporated. Appropriate encoding techniques ensure that these categories contribute effectively to predictive performance.

Feature reduction was approached in a structured yet exploratory manner. Initially, a correlation matrix was created to identify features that were highly correlated. These correlated pairs were tested iteratively by training models with and without the identified features, allowing assessment of their contribution to performance. The correlation matrix is presented in Figure 1. Although one can see that some feature clusters, like the ones quantifying delays are heavily correlated, they were found to all individually improve model performance. This is due to the features being available in different scenarios or at a different time on the prediction horizon. A clear example of this is the correlation between the actual delay features (prev_on/off-block_delay), which evidently follow each others trend, but still provide different insights as flight times can be influenced by external factors like weather or ATC routing. On the other hand, one can also see the "expected" departure and arrival delays are heavily correlated with the actual delays. Those become available at different times on the prediction horizon, and have a different reliability, making them still very useful for earlier predictions. Secondly, model-based feature importance scores, derived from permutation methods, were used to identify variables with limited predictive value. These were removed one by one while monitoring the model's accuracy. This approach helped to refine the input space while maintaining the model's performance and transparency.

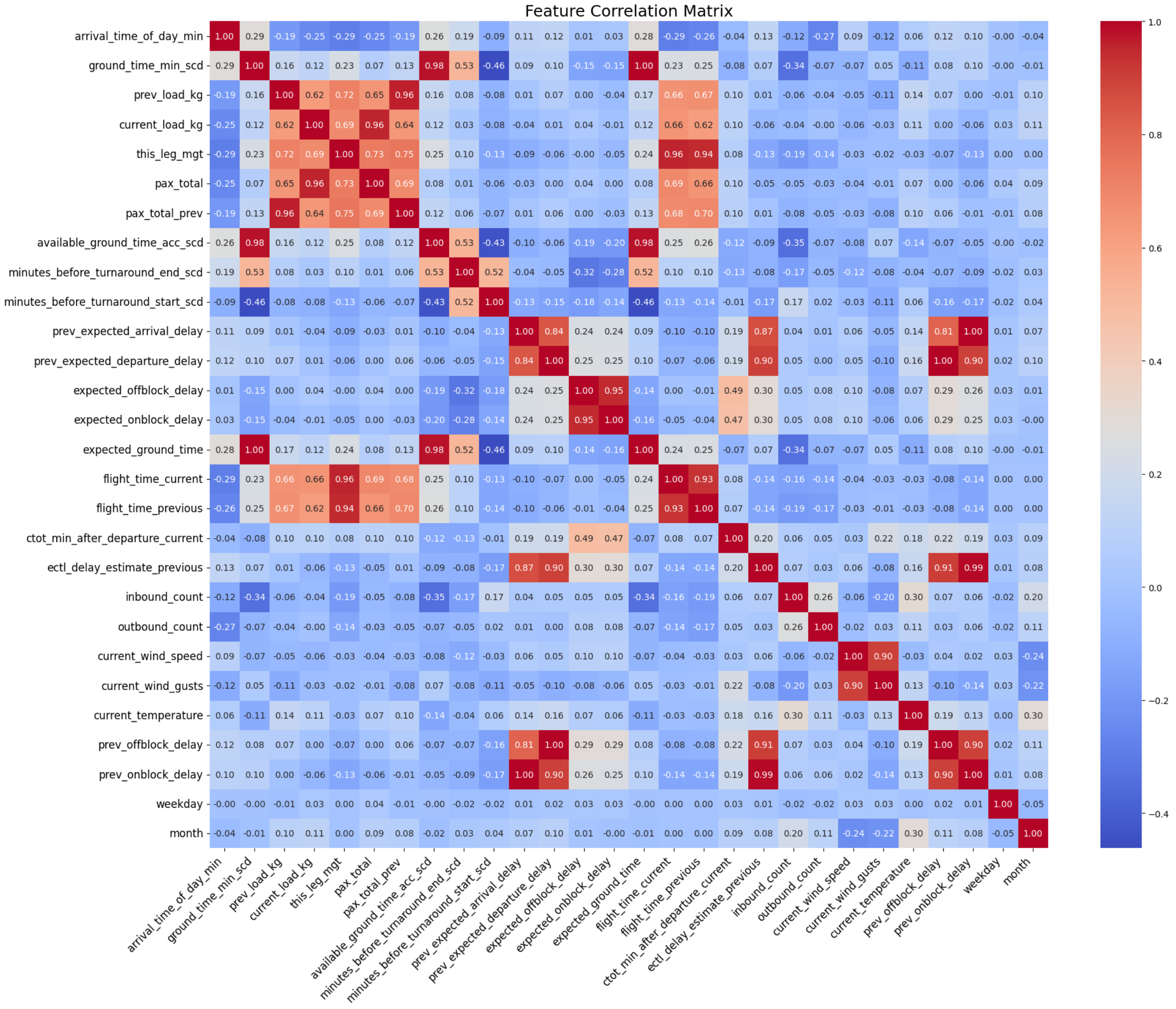


Figure 1. Training Features Correlation Matrix as from training data set

3.3 Model Selection & Splits

The research tested three types of supervised machine learning algorithms. Firstly, Random Forest, is made up of many decision trees [18] where each tree makes a prediction, and the model combines them to get a final result. This method helps avoid overfitting and can handle a mix of different features. Because Random Forests are easy to use and generally reliable, they are a good starting point when building and testing the model. Secondly, LightGBM is used for its speed and accuracy, benefiting from histogram-based splits and leaf-wise tree growth, which are efficient for large and high-dimensional datasets [19]. It is well adapted to the frequent updates and time constraints of operational prediction tasks. Thirdly, XGBoost extends this approach with more sophisticated boosting techniques, including second-order gradients and column subsampling [20]. Together, these models provide a flexible and scalable foundation for predicting turnaround times under dynamic, real-time conditions. Neural networks were not considered, as tree-based models have shown competitive performance in state-of-the-art research, while also offering better interpretability, which is an important factor for operational insights.

The next stage of the research explored how model performance could be improved by training separate models for different types of turnarounds. Three types of splits were considered. First, models were trained to focus on specific moments within the prediction horizon, particularly around operational milestones such as the previous flight's take-off and landing. Each model was trained on a subset of the data centred around these moments, allowing it to specialise in the data characteristics available at that point in time. Second, a split was introduced between long-haul and short-haul flights, as these operationally distinct categories follow different ground handling processes and durations. Finally, the models were trained based on whether a turnaround was "slacked" or "non-slacked". A turnaround was considered slacked when the scheduled time was 40 minutes more than the theoretical minimum turnaround time. The slack value was decided based on a sensitivity analysis, evaluating which threshold resulted in the most significant improvement. These model splits allowed a more targeted training process and provided valuable insight into how model accuracy and uncertainty calibration vary across operational contexts. It should be noted that for the horizon split, features that are generally not available early on the prediction horizon were excluded for training of that model. For the other splits the same set of features was used. Furthermore, it should be noted that all splits used the same hyperparameters.

3.4 Uncertainty Quantification

Quantile regression is employed to approximate the conditional distribution of the turnaround time. The model outputs the 5th, 50th, and 95th percentiles. The median (50th percentile) is retained as the point forecast, and the span between the 5th and 95th percentiles defines a central 90% prediction interval. The analysis especially focuses on how this interval contracts as the moment of interest approaches, rather than on its absolute width.

The models are trained with the quantile, or *pinball*, loss, defined in Equation 1 [21]. For a quantile level $q \in (0, 1)$ and an observation–prediction pair $(y, \hat{y})$, the loss is:

$$\ell_q(y, \hat{y}) = \begin{cases} q\,(y - \hat{y}), & y \geq \hat{y}, \\ (1 - q)\,(\hat{y} - y), & \text{otherwise.} \end{cases} \tag{1}$$

3.5 Hyper-parameter Tuning

For LightGBM and XGBoost hyperparameters are explored in two stages. An exhaustive grid search combined with stratified cross validation provides an initial map of the search space, after which Bayesian optimisation through Optuna concentrates the search on the most promising regions and stops unproductive trials early [22]. Model performance is measured with the Root Mean Equare Error (RMSE) on a held out validation fold, and training is stopped automatically when no further improvement is observed, which curbs overfitting. For the Random Forest the number of trees, maximum depth and node size are adjusted until the out of bag error estimate stabilises. The optimal configuration for each algorithm is stored together with the fitted preprocessing pipeline so that the exact model can be reproduced later.

The tuned hyperparameters for the three tree-based learners are summarised in Tables 2(a)–2(c). The resulting configurations indicate a preference for moderately deep trees and conservative learning rates.

Table 2. Hyperparameters for the three model types used.

(a) LightGBM parameters

Parameter	Value
$n_{\text{estimators}}$	200
learning_rate	0.0604
subsample	0.882
max_depth	16
colsample_bytree	0.808

(b) XGBoost parameters

Parameter	Value
$n_{\text{estimators}}$	159
learning_rate	0.0431
max_depth	9
subsample	0.885
colsample_bytree	0.832
tree_method	hist

(c) RF parameters

Parameter	Value
$n_{\text{estimators}}$	200
max_depth	16
min_samples_split	2
min_samples_leaf	1

3.6 Prediction Horizon and Evaluation Points

In the training data and local test data, the prediction horizon typically spans from hours before the turnaround to aircraft on-block. This timeline is thus different for every turnaround, depending on the duration of the previous flight and turnaround. Typically, the first data points are available around one to two hours before the previous departure. Evaluation focuses on overall performance over this horizon. However, the data is not uniformly distributed in time: the data density is higher around the turnaround phase, because of the more dynamic nature of this period compared to the previous ones. In turn, this means that predictions made closer to the off-block moment dominate the evaluation. Therefore, two specific prediction moments have been selected at which the results will be evaluated. These are the moment of the previous take-off and the moment of on-block of the previous flight, marking the start of the turnaround.

3.7 Performance Metrics

Accuracy is assessed using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). These performance metrics will be averaged over all individual prediction updates per flight to evaluate overall performance, and at two specific points on the prediction horizon, namely the previous take-off and the moment of the previous landing. Especially RMSE is the driving performance metric for this problem, as from a business perspective the biggest losses are encountered due to prediction that are very far off the true value. Furthermore, to capture the temporal evolution of the performance, the accuracy is also averaged per bucket of 20 minutes on the prediction horizon. Presenting the results in this manner anticipates the issue of the non-homogenous distribution of data over the prediction horizon, and shows how the model's performance evolves over time. A similar approach will be followed to evaluate uncertainty. Uncertainty is quantified by the central 90% predictive interval: the distance between the model's 5th and 95th conditional quantiles. Calibration is validated by comparing the empirical coverage of this interval with its nominal level; the closer the observed proportion of turnaround times inside the bounds is to 90%, the better the probabilistic forecasts are calibrated.

4. Experimental Setup

The experimental setup defines how the predictive models are developed, trained, and evaluated. It covers the integration of key data sources, the chronological train–test split, model configuration, and a live-like validation environment to ensure operational relevance.

4.1 Train–Test Partitioning

The dataset is partitioned chronologically to preserve temporal dependence and prevent the inadvertent use of future information during training. Records from January 2024 up to and including 31st March 2025 form the training set, whereas observations from 1st April 2025 to 15th June 2025 comprise the test set. The training set contains approximately 3.5 million data rows, covering around 66,000 turnarounds, while the test set comprises roughly 0.5 million rows corresponding to 12,000 turnarounds. This broad evaluation window captures both highly irregular and highly regular operational days, ensuring that no particular pattern exerts undue influence on the assessment of model performance.

4.2 Operational Validation Platform

While this paper focuses on offline model evaluation, the broader thesis project also includes a deployment phase to assess predictive performance in a live operational setting. At first instance, the model was tested locally. Data from multiple upstream systems was queried and preprocessed into a structured format suitable for training and evaluation. The framework was designed to allow for seamless switching between local testing and online deployment, with clearly separated logic for data processing and model inference.

All experiments and results presented in this paper are based on an offline test split of the queried dataset, which enables controlled evaluation under reproducible conditions. By contrast, the deployment phase, discussed in a separate thesis chapter, focuses on a continuously running system that schedules predictions for upcoming turnarounds up to 24 hours before arrival, updating them every 15 minutes based on the latest available data. The resulting predictions are stored and later compared to realised turnaround durations.

This operational configuration replicates key challenges encountered in a real-time environment, such as joining large and distributed databases at inference time, and being exposed to failure modes including data latency, data leakage, schema drift, and incomplete updates. By continuously backfilling the realised outcomes, the deployed setup enables near real-time monitoring of predictive accuracy, coverage, and calibration, and flags any deviation from the performance observed during offline validation. The design and evaluation of this operational validation campaign are documented in a dedicated chapter of the thesis.

5. Hypotheses

This section introduces 4 hypotheses on the results of this research, and outlines how each will be tested and presented in the Results in Section 6.

5.1 Hypothesis 1

The first part of the results evaluates three different models, each trained on the full training dataset as described in Section 4. This dataset spans the entire prediction horizon and is not partitioned in any way. As described in Subsection 6.1, model error metrics will be assessed from three perspectives. First, an overall view is presented by averaging the errors over the full prediction horizon and test set. Secondly, two specific points along the prediction horizon are used for evaluation: the previous departure and the start of the turnaround. For each turnaround, the predictions closest to either event are selected, and their errors are averaged. It is expected that the error metrics will decrease between these two moments, as the model benefits from increasingly rich real-time operational information, allowing it to adjust for any irregularities. Furthermore, since the data is denser closer to the turnaround start, the overall error is likely to reflect performance at that later point more strongly than at previous departure.

To mitigate this bias and to better understand how performance evolves along the prediction horizon, model accuracy is also assessed across complete horizons. Errors are averaged in twenty-minute evaluation buckets and plotted against two different reference points: minutes before turnaround start and minutes before previous departure. Although these time ranges may appear similar, they yield different perspectives, since each flight has a unique duration. In the training data, the average duration of a flight is 160 minutes, with short and long haul flights averaging 100 and 590 minutes respectively. Aligning by turnaround start or previous departure provides complementary interpretations. It is expected that the most significant improvements in performance will occur around the times when critical operational updates become available, particularly around previous departure, when the off-block time of the inbound aircraft becomes known, offering insight into schedule adherence.

5.2 Hypothesis 2

Uncertainty is quantified through the width of the central prediction interval and will be evaluated in the same way as the error metrics. It is hypothesised that this width will narrow in tandem with the reduction in error. The narrowing is expected to be most pronounced following the previous departure, when the actual off-block time becomes available, and again shortly before stand arrival, when slot allocations and load figures are typically updated. These patterns are examined in Subsection 6.2, which presents both the overall and time-dependent evolution of the 90% central prediction interval. Additionally, the coverage of these prediction intervals will be evaluated, to validate that the decrease in interval width actually corresponds to a reduction in uncertainty and not an increase in outliers.

5.3 Hypothesis 3

With regard to feature relevance, available ground time is expected to be the dominant driver of the median (50th quantile) prediction throughout the horizon, consistent with prior findings. At earlier horizons, static structural features such as aircraft type, destination, scheduled turnaround time, and slack are likely to rank highly. As the turnaround approaches, dynamic indicators, such as delay propagation signals and load-related variables, are expected to gain prominence. A similar shift is anticipated for the extreme quantiles: schedule-driven slack is expected to dominate the lower bound, whereas the upper tail is likely influenced more by cumulative delay signals. These expectations are tested in Subsection 6.3, which presents gain-based importance scores and Shapley value summaries for the median and extreme quantiles, at both early and late horizons.

5.4 Hypothesis 4:

Lastly, in addition to the baseline models, the results also assess the effect of partitioning the training data and training split models accordingly. These dedicated models are expected to offer improved error metrics and reduced uncertainty. Partitioning the training set by prediction horizon, by long-haul versus short-haul operations, or by the presence of substantial slack is presumed to reduce the variability of the target variable within each subset, thereby improving predictive performance. These hypotheses are tested by comparing the baseline results with the performance of the split models in terms of both point-wise error and interval width, as presented in Subsection 6.4.

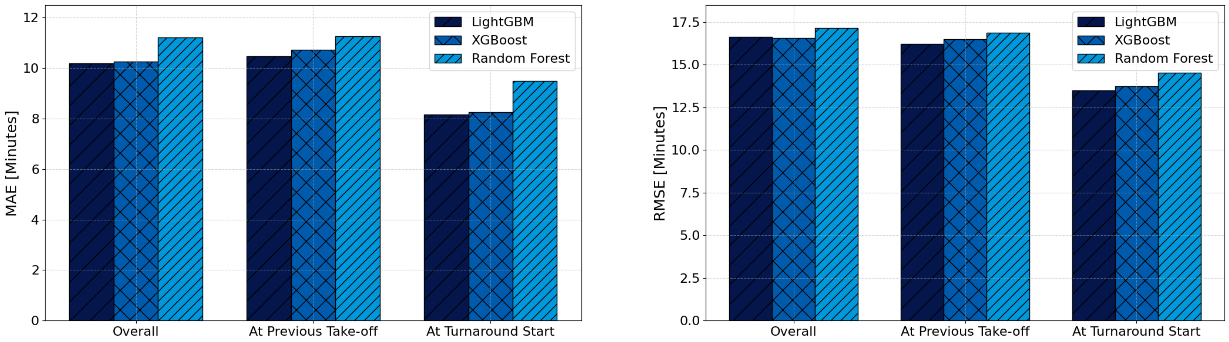
6. Results

This section presents the performance of the proposed models in predicting turnaround time distributions. The findings are structured in accordance with the expectations outlined in Section 5.

Subsection 6.1 first reviews the accuracy of the three candidate algorithms, which are trained with the complete non-split training data, across the complete prediction horizon and at two operationally relevant instants; previous departure and start of the turnaround. The analysis then turns to predictive uncertainty: Subsection 6.2 traces the width of the central prediction interval (5th to 95th quantile) and explains how it tightens as additional information becomes available. Additionally, the coverage of these prediction intervals will be evaluated, to validate that the decrease in interval width actually corresponds to a reduction in uncertainty and not an increase in outliers. Attention subsequently shifts to explanatory factors in Subsection 6.3, where gain scores and Shapley values reveal how the relative influence of structural and dynamic variables evolves with horizon and quantile. The section concludes with Subsection 6.4, which tests the change in performance when specialised models are trained on partitioned data and assesses any gains in either accuracy or uncertainty when compared with the baseline models, trained with the non-partitioned training set.

6.1 Model Performance

The performance comparison of the three selected modelling techniques, as introduced in Section 3, is shown in Figures 2(a) and 2(b). This analysis covers both the mean absolute error (MAE) and the root mean square error (RMSE), and considers models trained on the full dataset, which spans the entire prediction horizon.



(a) MAE

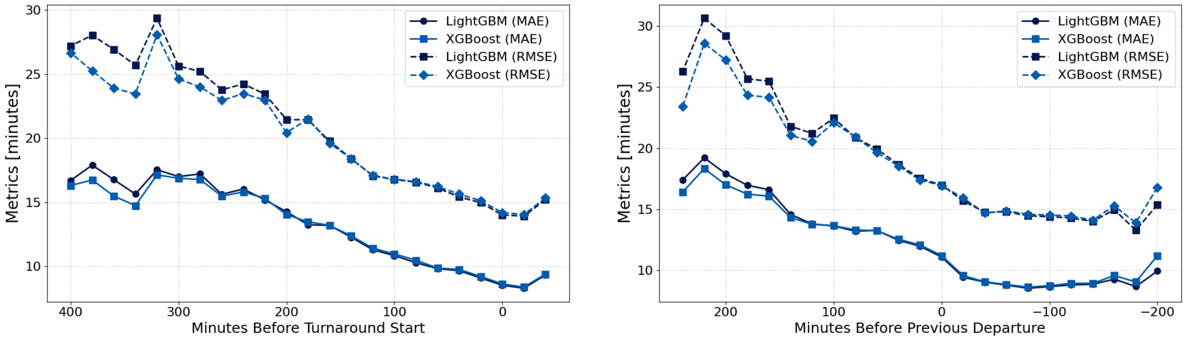
(b) RMSE

Figure 2. Model performance comparison in terms of MAE and RMSE across three evaluation scenarios.

Three different evaluation scenarios are considered to start. The first, denoted as "Overall", includes all test data points across the full range of prediction horizons and yields a MAE of approximately 10 minutes for both XGBoost and LightGBM, whereas Random Forest records about 11 minutes; the corresponding RMSE is 16 minutes for XGBoost and LightGBM and 17 minutes for Random Forest. However, these updates are not uniformly distributed in time: the data density is higher around the turnaround phase, because of the more dynamic nature of this period compared with the preceding stages. In turn, this means that predictions made closer to the off-block moment dominate the evaluation. As predictive accuracy naturally increases with proximity to the turnaround moment, the overall performance metric gives a biased view. For this reason, two additional evaluation moments are defined: the moment of the previous take-off and the moment of the turnaround start, for which the error metrics are also presented.

It can be seen that both LightGBM and XGBoost consistently outperform Random Forest, achieving lower error metrics across all evaluation moments. The performance of these two boosting models is similar at each horizon, with LightGBM having a slight edge. Furthermore, the largest errors are observed at the moment of the previous take-off, when only limited operational information is available. Based on this finding, subsequent modelling efforts have focused exclusively on LightGBM and XGBoost. Random Forest was excluded from further experimentation, given its consistently inferior performance in both MAE and RMSE across all evaluated scenarios.

To complement the aggregated view, Figures 3(a) and 3(b) present the performance metrics over the relevant prediction horizon, plotted against two distinct reference timestamps: minutes before turnaround start and minutes before previous take off.



(a) From turnaround start

(b) From previous take-off

Figure 3. Evolution of model performance over the prediction horizon, measured from both turnaround start and previous take-off.

When using the turnaround start as reference (Figure 3(a)), the evaluation reflects how model performance evolves with proximity to the ground operations phase. A steady decline in both MAE and RMSE can be observed as the turnaround moment approaches, highlighting the increased availability and quality of information closer to ground events. LightGBM and XGBoost follow nearly identical trajectories. It should be noted that the average flight time is around 160 minutes, resulting in the test data earlier on the time horizon being very sparse. Therefore, the performance seems irregular in the 400 to 200 minutes before turnaround range. Additionally, another small increase can be noted after the start of the turnaround. During this time the data is very dense, but sometimes not very consistent towards the end of the turnaround as target departure times are continuously updated. It is assumed that this leads to the decreased performance.

Alternatively, using the previous departure as anchor provides a view on how performance develops before and throughout the inbound flight. As data becomes available during the airborne phase, the models are able to incorporate more contextual factors, leading to steadily improving accuracy. It can be observed that the strongest drop happens shortly after the departure of the previous flight, showing that the actual departure time of the previous flight helps refine the prediction significantly. This indicates that once the previous flight is airborne and no longer taxiing, a better prediction can be made. The performance stabilises throughout the previous flight, where the most relevant features are already available, and there are thus no major new features to adjust the prediction. This perspective illustrates how early predictions can already attain a reasonable level of accuracy, with errors dropping once the aircraft is airborne, but not improving much afterwards as the most important operational context is known at this point. Furthermore, similarly to the turnaround horizon, it can be seen that performance of early and late predictions seem to exhibit worse performance. This is assumed to be for the same reason as before.

6.2 Uncertainty Quantification

Predictive uncertainty is examined in two complementary steps. First, the width of the central prediction interval ($Q_{0.05}$–$Q_{0.95}$) is taken as a proxy for uncertainty and compared at two operational milestones: immediately after the previous departure and at the start of the next turnaround. The same metric is then tracked across prediction horizons anchored to each milestone in order to reveal how uncertainty evolves as additional information becomes available. Second, interval coverage is assessed by confronting the nominal 90% band with the empirical proportion of turnaround times that fall inside it. This validation confirms whether narrower intervals truly reflect reduced uncertainty rather than a proliferation of outliers, and it exposes any systematic tendency of the models to under- or over-predict.

The predictive uncertainty associated with the two shortlisted models is summarised in Figure 4. Each bar represents the width of the prediction interval at two evaluation moments: shortly after the previous take-off and just before the subsequent landing. For both LightGBM and XGBoost, interval widths remain broadly stable over time, indicating limited refinement of the predicted distribution as more information becomes available. Only a modest reduction is observed for LightGBM in the lower tail between the two prediction moments, indicating that the model benefits somewhat from the additional information available closer to landing. Furthermore, LightGBM produces consistently narrower intervals across all quantile spans, suggesting greater confidence in its predictions.

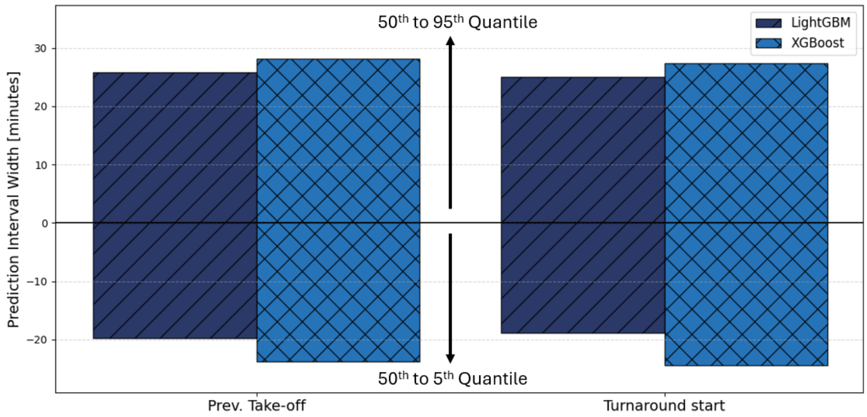
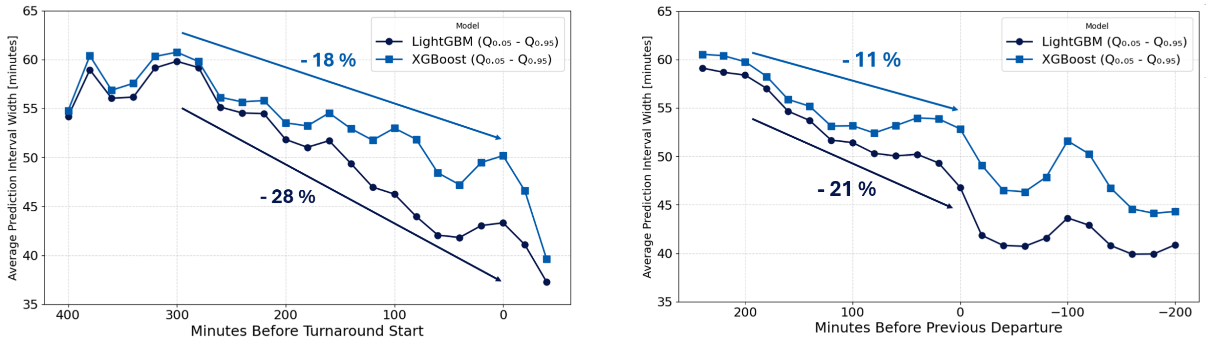


Figure 4. Prediction interval widths for LightGBM and XGBoost at two evaluation moments. Intervals shown for full (5th–95th), lower (5th–50th), and upper (50th–95th) quantile widths.

This behaviour is further clarified by the temporal evolution illustrated in Figures 5(a) and 5(b). The curves trace the mean width of the $Q_{0.05}$–$Q_{0.95}$ interval for successive 20 minute evaluation buckets derived from the test set. When the countdown is expressed relative to the start of the turnaround (Figure 5(a)) the interval oscillates gently until roughly 300 minutes before gate arrival. Thereafter, a pronounced and sustained contraction begins. Between 300 minutes before turnaround start and the start itself the LightGBM interval width falls by 28%, whereas the XGBoost counterpart contracts by 18%. The turning point coincides with the moment at which, on average, the preceding flight departs; from that stage a reliable estimate of the inbound arrival time becomes available and, as a result, the size of the ground window can be assessed with greater accuracy. During the final hour before the aircraft reaches the stand the curves level out because usually no crucial operational updates come in that would sharpen the forecast.

The complementary perspective that fixes the previous departure as the temporal anchor (Figure 5(b)) produces a consistent picture. One should note that the steepest decline is centred on the departure itself. Furthermore, between 200 minutes before take-off of the previous leg and the event the LightGBM width diminishes by 21% and the XGBoost value by 11%. Once the aircraft is airborn the rate of reduction moderates, and even a slight temporary increase can be seen around 100 minutes after previous departure. Additionally, a small increase in uncertainty can be seen around 100 minutes after previous departure.

It is assumed that, similarly to the error metrics, this is due a high number of operational updates coming in that are not always as consistent, increasing uncertainty.



(a) From turnaround start

(b) From previous take-off

Figure 5. Evolution of predictive uncertainty, defined as the average $Q_{0.05}$–$Q_{0.95}$ prediction interval width, over the forecast horizon, measured from both turnaround start and previous departure.

To continue, calibration was assessed by comparing the empirical coverage of the central 90% predictive interval with its nominal level. Table 3 reports the proportion of turnaround times whose true values fell inside the full $Q_{0.05}$–$Q_{0.95}$ band and, for diagnostic insight, within each of its halves. Across the entire horizon, coverage of the full band approached the target yet remained slightly below it. LightGBM rose from 87% overall to 89% at the start of the turnaround, showing that the additional information not only narrows the interval but also mitigates under-coverage, thereby evidencing a genuine reduction in uncertainty. XGBoost hovered close to 89% at all horizons.

The half-interval decomposition reveals a mild asymmetry: both models allocate more probability mass below the median (the 5th–50th-quantile band) than above it, and the imbalance is greater for LightGBM. Because a larger share of true values lies in the lower half of the predictive distribution, both models tend to over-predict turnaround time, with the tendency slightly more pronounced in LightGBM.

Table 3. Empirical coverage (%) of the central 90% prediction interval and its two halves.

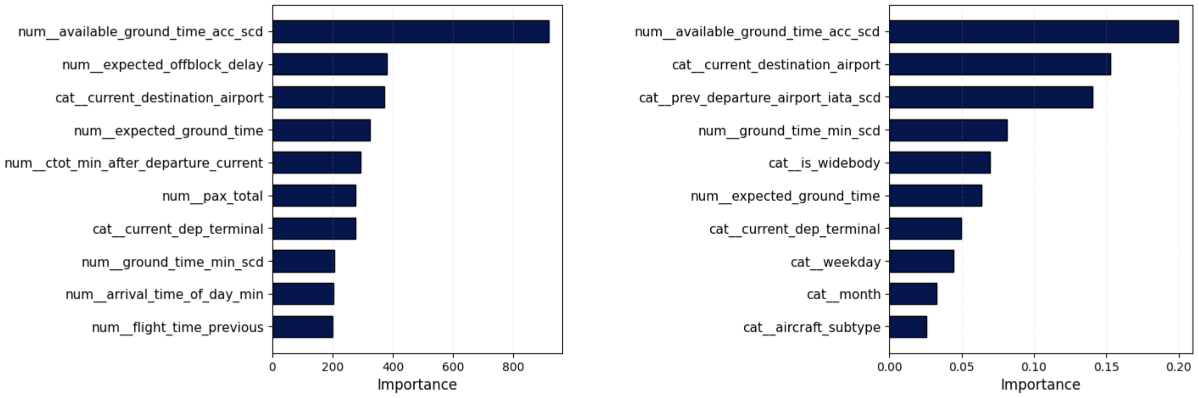
	Overall	Previous Take-off	Turnaround Start
LightGBM			
$Q_{0.05}$–$Q_{0.95}$	86.95%	88.04%	89.21%
$Q_{0.05}$–$Q_{0.50}$	48.70%	46.66%	51.86%
$Q_{0.50}$–$Q_{0.95}$	38.26%	41.41%	37.35%
XGBoost			
$Q_{0.05}$–$Q_{0.95}$	86.96%	88.33%	88.89%
$Q_{0.05}$–$Q_{0.50}$	47.90%	45.30%	50.03%
$Q_{0.50}$–$Q_{0.95}$	39.07%	43.07%	38.89%

6.3 Feature Importance

Understanding which features drive predicted turnaround time supports both model assessment and operational insight. Gain based rankings from LightGBM and XGBoost are presented first; because these scores are aggregated during training they summarise importance across the full prediction horizon. Additionally, the Shapley values for LightGBM and XGBoost based on the full "overall" prediction horizon will be presented. Next, LightGBM Shapley values are inspected at the previous departure and at the start of turnaround to reveal how influence shifts as operational updates come in. Finally, Shapley values at the 5th and 95th percentiles identify the variables that drive the lower and upper bound of the prediction interval. It should be noted that only the Shapley analyses can work with partitioned test sets for comparison; the gain analysis cannot be partitioned and therefore always refer to the full horizon.

6.3.1 Comparing Models

The gain based bars in Figures 6(a) and 6(b) reveal a clear consensus between the two gradient boosted ensembles. Both assign largest importance to *available ground time*, underscoring the central role of turnaround slack in shaping predictive performance. Beyond this feature, the models diverge in the variables to which they attribute secondary influence. LightGBM places greater weight on the evolving state of the rotation, with the most recent estimate of off-block delay, the Calculated Take Off Time offset and the total passenger load all ranked within the top six contributors. XGBoost, by contrast, elevates network and fleet descriptors such as the current destination airport, the previous departure airport and a wide body flag, while temporal markers such as weekday and month also feature prominently. These distinctions suggest that XGBoost draws more of its explanatory power from structural scheduling patterns, whereas LightGBM captures urgent operational signals that arise on the day.

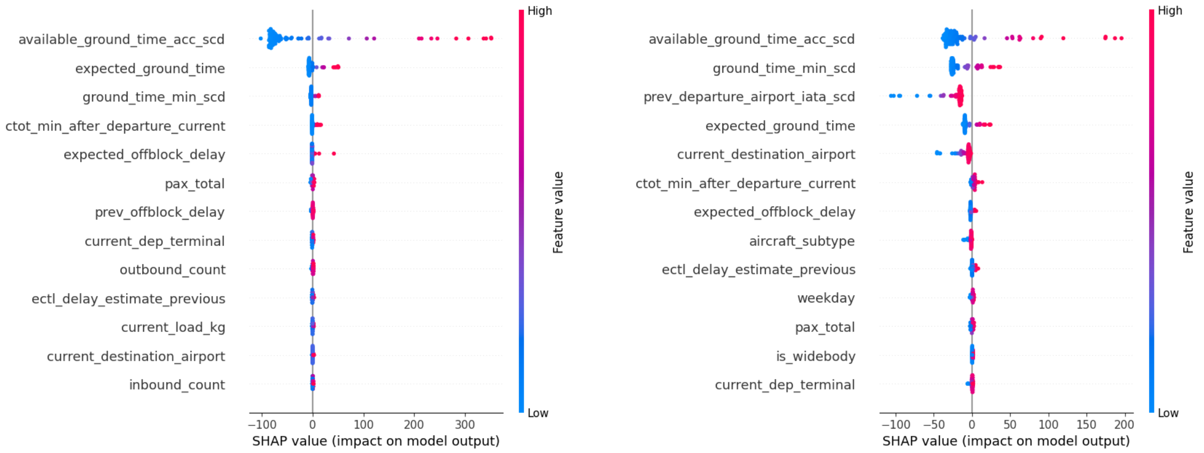


(a) LightGBM

(b) XGBoost

Figure 6. Comparison of gain-based feature importances for the LightGBM and XGBoost models.

The interpretation derived from the gain graphs is corroborated by the SHAP [23] summary plots in Figures 7(a) and 7(b). In both models the largest absolute Shapley values are again associated with turnaround slack variables, confirming that deviations in the planned buffer remain the dominant driver of the conditional median. The LightGBM plot reinforces the heightened sensitivity to real-time delay information and passenger volumes, whereas the XGBoost plot reiterates the importance of network context and aircraft characteristics. This could be explained by the fact that origin and departure airport also partially serve as a proxy for average booking figures in terms of passengers and cargo. However, for the XGBoost model, the Shapley values do show a more nuanced view then the gain graphs, with more numerical and dynamic features ranking higher. Still, it could be said that XGBoost seems to focus more on proxies, while LightGBM captures the real influence better. Quantitatively, the peak absolute Shapley impact approaches 300 minutes for LightGBM, which is appreciably higher than the 200 minute ceiling observed for XGBoost. This indicates that LightGBM predictions react more strongly to unusually large buffers, while XGBoost spreads influence more evenly across a broader feature set.

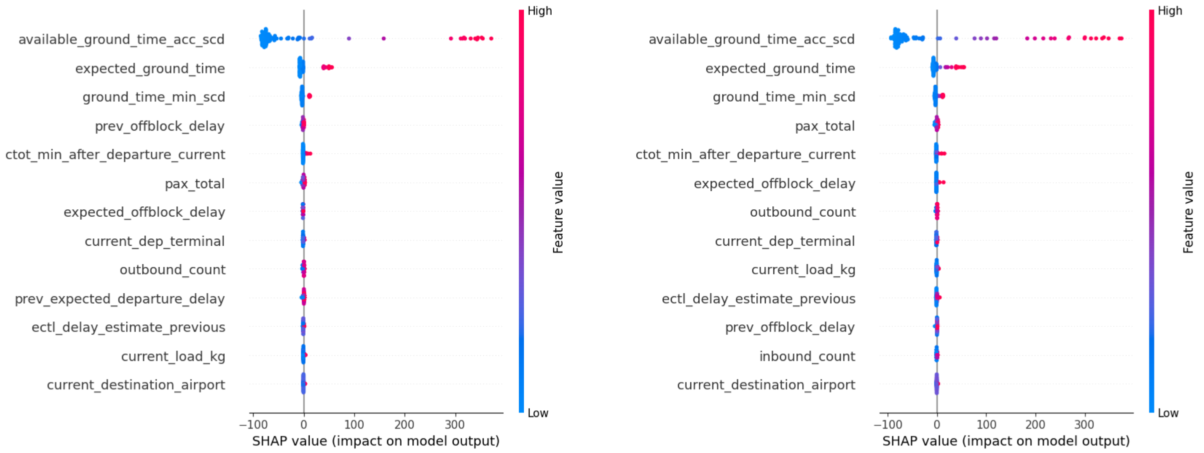


(a) LightGBM (b) XGBoost

Figure 7. SHAP summary plots at the fiftieth percentile for both models.

6.3.2 Comparing Prediction Horizons

A complementary experiment isolates the LightGBM model and contrasts its feature ranking when predictions are produced at two distinct moments of the rotation: immediately after the previous take off and at the beginning of turnaround. Figure 8(a) and Figure 8(b) show that the leading feature, available ground time, remains dominant, yet the secondary drivers shift markedly: At the previous take off moment the model assigns greater importance to delay updates that describe the incoming leg. The variables previous off block delay and previous expected departure delay rise to prominence, alongside the current CTOT offset, indicating that the prediction is anchored in the operational state of the arriving flight. Passenger and load metrics play a lesser role, as the forthcoming boarding process is still remote in time. However, by the start of turnaround, the emphasis reverses. Load related attributes, most notably total number of passengers and current load in kilograms, climb the ranking, while delay updates tied to the completed sector recede. The model therefore shifts focus from upstream uncertainties to the ground activities that will govern the remaining turnaround phase. This temporal reweighing underscores the adaptability of the LightGBM explainer and confirms that operational context determines the explanatory power of individual features at the median.

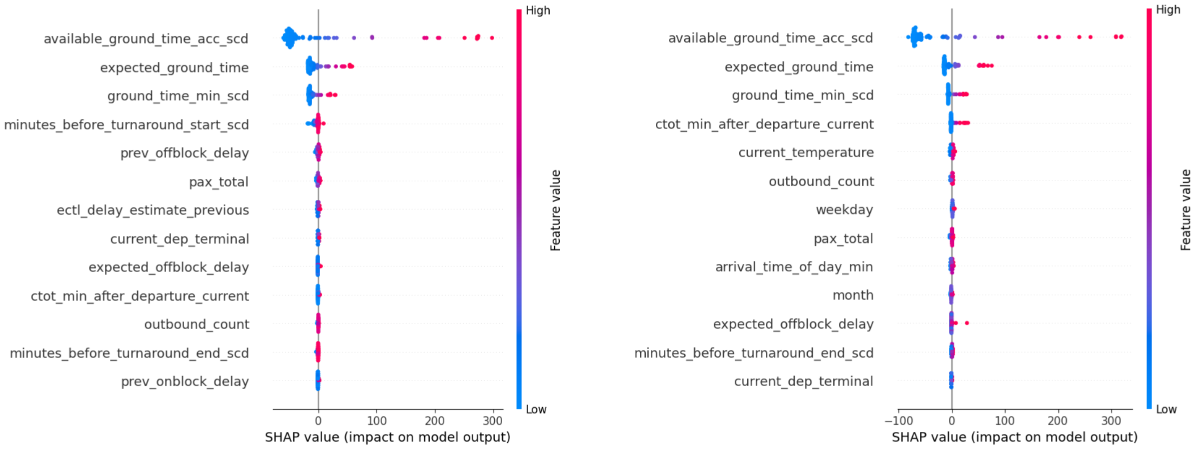


(a) At Previous Take-Off (b) At Turnaround Start

Figure 8. LightGBM SHAP summary plots at the fiftieth percentile for two prediction moments.

6.3.3 Comparing Quantiles

Finally, an inspection of the extreme quantiles reveals a consistent shift in emphasis. When the model targets the 5th percentile it relies almost exclusively on the available ground time and the scheduled duration on stand, indicating that punctual turnarounds occur only when a generous buffer is already embedded in the plan. At the 95th percentile the hierarchy changes: variables that measure accumulated disruption, most notably the previous off block delay and the current CTOT offset, gain prominence, whereas passenger and load factors recede. The contrast confirms that early departures remain schedule driven, whereas chances for prolonged turnarounds are governed by cascading delay signals.



(a) 5th percentile

(b) 95th percentile

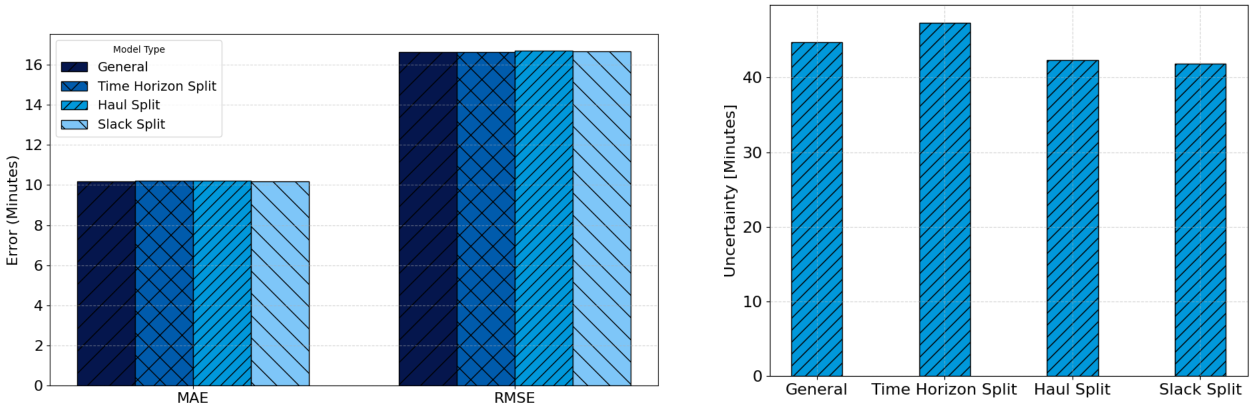
Figure 9. LightGBM SHAP summary plots for the 5th and 95th percentiles.

6.4 Split models

Figure 10(a) presents the MAE and the RMSE obtained for the baseline general model and for the three dedicated models trained on partitioned data: a separation by prediction horizon, a separation between long haul and short haul flights, and a separation according to the presence of operational slack in the schedule. The average error stays slightly above 10 minutes for every configuration, indicating that the split strategies do not provide a systematic reduction in point wise error. Also the root mean squared error remains virtually identical across the four configurations, fluctuating above 16 minutes. This negligible variation indicates that the data partitions do not materially influence the occurrence of larger residuals.

Figure 10(b) shows the corresponding predictive uncertainty, expressed as the interquantile width produced by the LightGBM model. The general model yields an average spread of approximately 45 minutes. Partitioning by horizon leads to the widest distribution, just over 47 minutes, whereas the haul and slack partitions narrow the spread to roughly 42 minutes. Hence, although point wise accuracy is unchanged, the partitioned models modestly reduce the predicted uncertainty, with the slack based separation achieving the lowest average width. Coverage was found to remain consistent between general and split models.

The temporal evolution of the predictive intervals reinforces the conclusions drawn from the aggregate statistics. Figures 11(a) and 11(b) plot the mean width of the central 90 % prediction interval against, respectively, the time relative to turnaround start and the time relative to the previous departure. All configurations exhibit the expected step change immediately after the preceding flight leaves the gate, reflecting the arrival of a reliable off-block timestamp. Thereafter the curves diverge. The interval generated by the general model flattens ahead of turnaround commencement



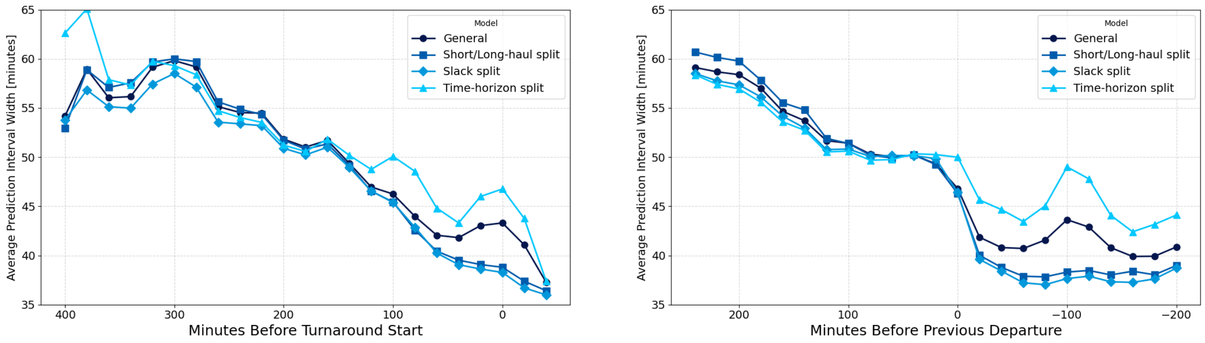
(a) MAE and RMSE

(b) Predictive uncertainty

Figure 10. Performance and uncertainty for the baseline model and three split configurations.

and shows no further contraction. In contrast, both the haul specific and the slack specific models sustain a steady decline, reducing the average width further. The likely explanation is a reduction in scenario diversity: when the model is exposed only to short turnarounds or only to generous ones the conditional spread of the target variable contracts. In practice, it must be evaluated whether the modest benefit in confidence justifies the additional maintenance overhead of multiple specialised models. This shows that the haul and slack models keep leveraging the new operational information, whereas the general model uncertainty stagnates. The horizon based model remains the most cautious and its line fluctuates markedly.

The picture is subtler when the timeline is expressed relative to the preceding departure. Although the haul and slack curves sit consistently below the general line, they also level off at broadly the same moment. Only the general and the horizon-split models display a brief uptick near 100 minutes after previous departure. Hence, on this axis, the benefit of the dedicated splits lies in a lower plateau after departure rather than in a longer downward trend.



(a) From turnaround start

(b) From previous take-off

Figure 11. Average $Q_{0.05}$–$Q_{0.95}$ prediction interval widths over time for the baseline and split models.

7. Performance Over Time and Use-Case Analysis

This chapter further reviews the predictive performance of the LightGBM turnaround model. First, in Subsection 7.1, it quantifies aggregate gains over a dynamic rule-based baseline, tracing weekly error profiles across the test period. Secondly, in Subsection 7.2 it dissects two representative cases to illustrate how the model integrates real-time delay intelligence and slot constraints, thereby translating operational signals into refined, situation-specific forecasts.

7.1 Performance validation

The timeline presented in Figure 12 positions the LightGBM scores alongside the error realised by the *Expected Ground Time* baseline, a simple dynamic estimate that reflects only the latest schedule and delay inputs.

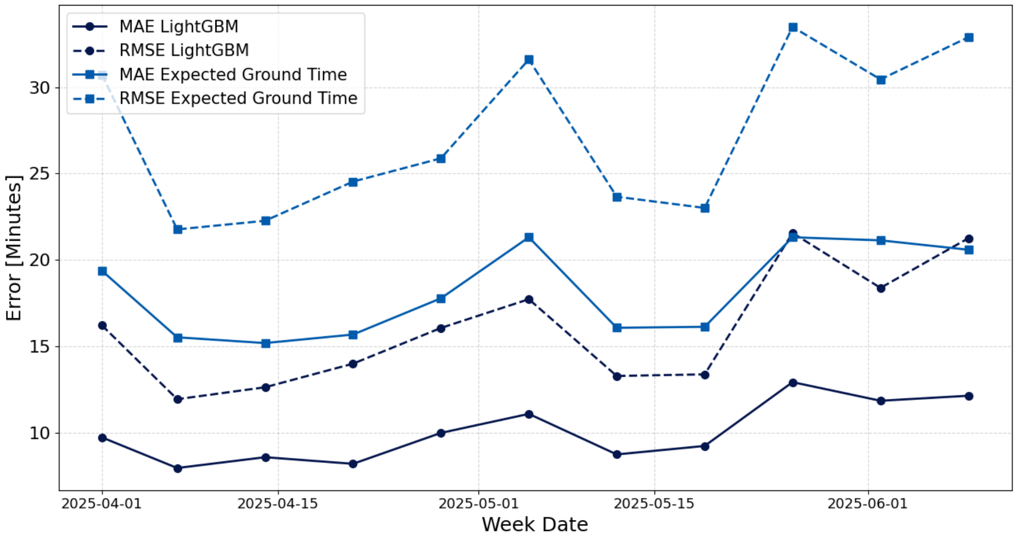


Figure 12. Weekly evolution of the mean absolute error (MAE) and root mean square error (RMSE) for the LightGBM model compared with the Expected Ground Time baseline on the test period.

The gap between the two curves is persistent and averages roughly six minutes for MAE and about ten minutes for RMSE, which demonstrates that a learning model integrating a richer set of operational drivers outperforms an updated rule-based forecast.

The figure also reveals pronounced weekly volatility. Periods characterised by orderly operations, such as the early weeks of April, coincide with the lowest errors, whereas holiday peaks around Easter and the subsequent surge in slot restrictions drive the largest residuals. This pattern confirms that the present feature set does not fully capture the mechanisms that govern outlier days; when disruption happens, the explanatory variables seem to provide an incomplete picture. It also confirms that it is difficult to asses a model's performance based on absolute error values. Performance is highly variable, and depends on the regularity of the operations.

7.2 Prediction Examples

This subsection discusses two specific prediction examples, showing the operational understanding of the model.

Example A: delay-driven compression:

As shown in Figure 13, at the start of the horizon, there is already information available on the expected arrival delay of the inbound flight. The model therefore centres its median forecast about 20 minutes below the scheduled turnaround time. Right before the inbound leg becomes airborne, the available turnaround shortens abruptly to about 70 minutes and the entire distribution shifts downward, as the flight left with even more delay than foreseen earlier. The contraction of uncertainty is immediate because an important feature, the actual departure, has been resolved.

Another steep rise in expected arrival delay about 40 minutes later compresses the prediction further. The 5th percentile is now constrained by the airline minimum, the median stabilises near 65 minutes, and the upper tail falls beneath ninety minutes. Minor adjustments in slot-driven departure constraints introduce only small ripples thereafter, so the prediction remains tightly aligned with the eventual outcome of around 60 minutes. The figure shows how real-time delay intelligence can erode the comfort of a generous schedule and how predictive confidence increases as the tactical horizon shrinks, and important information comes in.

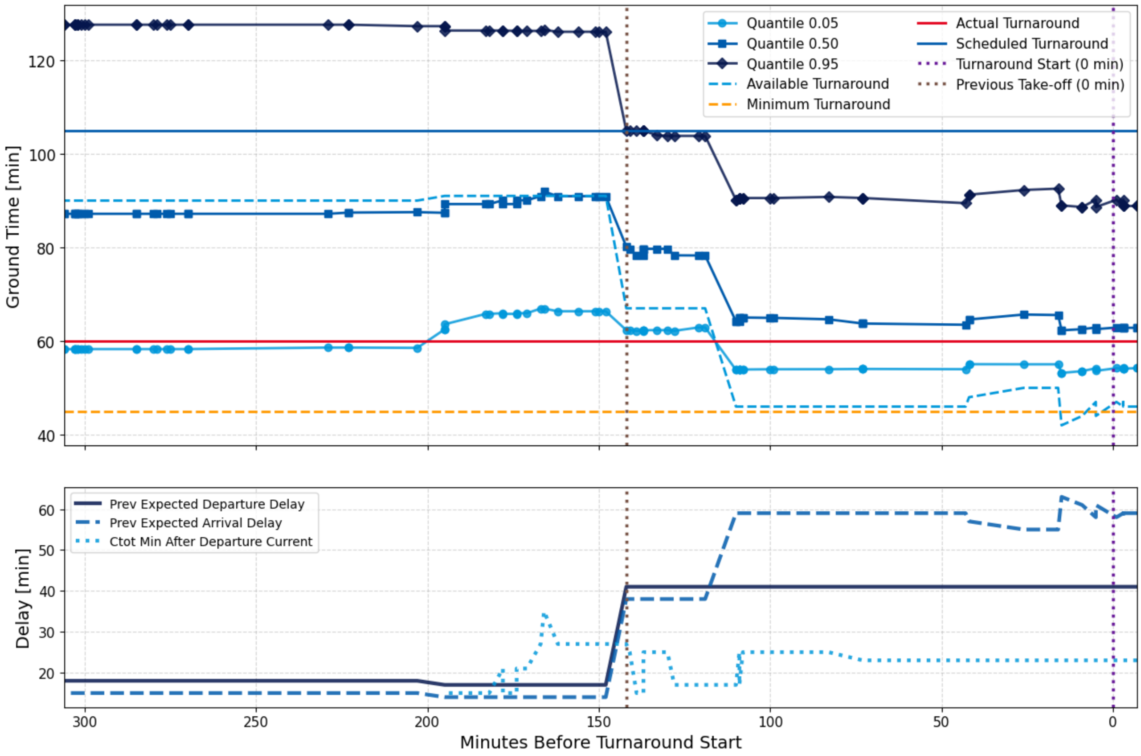


Figure 13. Evolution of the predicted turnaround distribution for Example A with important delay features.

Example B: slot-induced extension:

Next, on Figure 14, it can be seen how during the initial horizon, almost 4 hours before ground operations, the forecast mirrors the scheduled allowance of 70 minutes and exhibits limited spread. The picture changes when an air-traffic slot is issued for the onward sector, stipulating that the aircraft may not leave Brussels until at least one 110 minutes after its scheduled off-block. That constraint inflates the feasible window, lifting the 95th percentile to around 145 minutes while nudging the median only modestly upward. The fact that the median is only slightly adjusted, shows that the model understands that this CTOT information is still unreliable, and that there are high chances of still getting a better slot assigned as the turnaround approaches.

Once the inbound flight departs, both the arrival time and the slot coordinator's revisions act to tighten uncertainty. Successive reductions in the earliest permitted off-block time allow the available turnaround to edge downward; the 5th percentile approaches the airline minimum and the upper tail descends in a series steps. By the time the aircraft reaches the stand, the model has converged on a ground time of 80 minutes, exactly what the operation ultimately required. The example demonstrates that a departure slot can extend the feasible turnaround far beyond what service tasks alone would dictate and that the model incorporates such constraints as soon as they appear, but also understands that it is not a given the new CTOT will be the actual time of departure of the outbound flight, and refines its estimates as certainty improves.

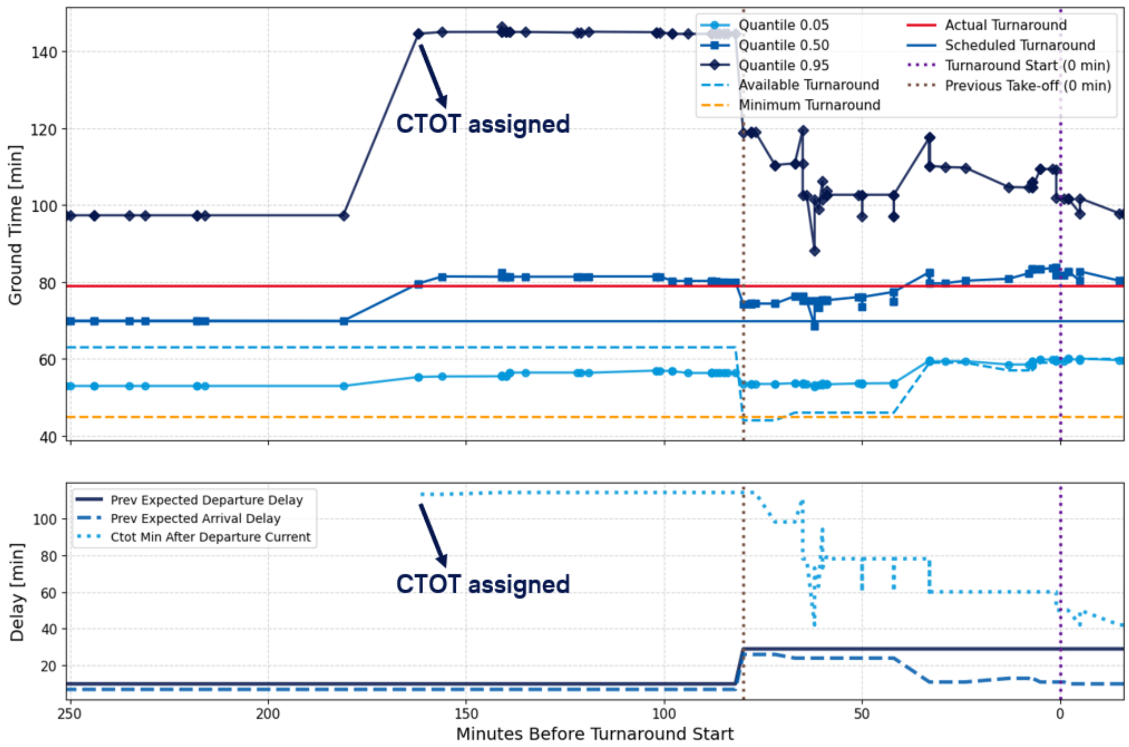


Figure 14. Evolution of the predicted turnaround distribution for Example B with important delay features.

8. Discussion

This section interprets the findings in light of the study's central objective, namely to determine whether a continuously updated probabilistic model can capture aircraft turnaround time across the entire tactical horizon while providing operationally meaningful uncertainty information. First, Subsection 8.1 covers the main insights, positions the results against current state-of-the-art work, and discusses the relevance. Next, Subsection 8.2 revisits the hypotheses set out earlier and assesses the extent to which they are supported by the results. Finally, Subsection 8.3 reflects on the study's limitations and outlines future research potential.

8.1 Main Observations

From an error perspective, the models developed in this work demonstrate performance that is in line with state-of-the-art research. For instance, the departure delay prediction model from prior research, which had an MAE of 8.3 minutes 90 minutes before take-off, improved to 7.4 minutes 15 minutes before take-off. While our prediction horizon is broader and starts earlier in the turnaround process, the performance of our turnaround time predictions aligns well with this departure delay model. Specifically, at the time of the previous take-off, our turnaround predictions had an error of 10 minutes, which reduced to 8 minutes by the start of the turnaround. However, it should be noted that a direct comparison is not feasible, as predictive performance is highly sensitive to context and methodology, as discussed in Subsection 7.1.

Furthermore, taking all the results together, it is clear that the model is able to capture how uncertainty evolves over time and effectively integrates new information to refine its predictions. This is especially true for LightGBM, which allocated more importance to dynamic features, such as delays, to reduce uncertainty as the turnaround approaches. This feature importance highlights the model's ability to adapt and make more accurate decisions as new data becomes available, leading to increasingly confident predictions closer to the operational event.

This dynamic updating of uncertainty is highly valuable in real-world operational contexts. As illustrated by the prediction examples in Subsection 7.2, the model's uncertainty closely aligns with the variability seen in operational situations, providing actionable insights for operational controllers. The evolving uncertainty, which narrows as the turnaround progresses, gives operations teams the information they need to respond to disruptions more effectively, adjusting their strategies with a clear understanding of both the predicted outcome and its associated risk. Furthermore, beyond aiding operational controllers, the complete output of the model offers significant potential for integration into operations optimisation engines used by the airline. With both the predicted turnaround time and its uncertainty, these optimisation engines can adjust the risk cost or weight of potential solutions. This enables the identification of the optimal solution to a disruption.

8.2 Hypotheses Validation

The results largely support the hypotheses formulated in Section 5. As anticipated by Subsection 5.1, model accuracy improved as the prediction horizon shortened, confirming that real-time updates significantly enhance predictive precision. The clearest gains occurred shortly after the inbound flight became airborne, when crucial timing information about the rotation is resolved. This validates the assumption that performance evolves meaningfully across the horizon and that early predictions, while informative, are progressively refined as operations unfold. However, it is also important to note that predictions occasionally became less stable, with errors increasing towards the end due to the volatility of the available information.

The expected reduction in predictive uncertainty from Subsection 5.2 was also confirmed. LightGBM in particular demonstrated a consistent narrowing of its prediction intervals as more data became available, indicating that the model is capable of translating information updates into a quantifiable decrease in uncertainty. Crucially, this narrowing was shown to correspond to actual improvements in coverage, rather than merely reflecting overconfidence, supporting the hypothesis that uncertainty metrics remain well-calibrated across time.

Feature importance trends, discussed in Subsection 5.3, confirmed the central role of turnaround slack, with a dynamic shift in explanatory variables as the turnaround approached. The model relied more heavily on structural features at earlier horizons, while operational and delay-related inputs took over at later stages. This adaptive behaviour matches the hypothesised temporal reweighting of features and underscores the model's responsiveness to changing operational context. Especially for LightGBM it was observed that greater weight was put on dynamic predictors such as current delay offsets. This distinction likely underpins the sharper reduction in LightGBM uncertainty: a model that values dynamic inputs can re-shape its conditional distribution more responsively than one that is anchored to static context.

The hypothesis that split models would outperform the general model from Subsection 5.4, was only partially validated. While point-wise accuracy remained largely unchanged, some partitions, particularly by haul and slack, achieved a modest reduction in predictive uncertainty. This suggests that targeted specialisation can be beneficial in certain configurations, though the operational value must be weighed against the added complexity of model maintenance.

8.3 Limitations and Future Work

While the proposed framework demonstrates strong overall performance, several limitations remain. Most notably, as discussed in Section 7, predictive accuracy varied considerably over time. In particular, the model struggled to fully respond to sudden schedule disruptions, suggesting that not all drivers of operational volatility are adequately captured by the current feature set. This points to a gap in how irregularities propagate through the system and influence turnaround outcomes. Next, another limitation lies in the scope of the dataset. The model was developed and tested using data from a single airline and focused exclusively on its hub operations. As a result, it covers only approximately half of the airline's total turnarounds, and its applicability to outstation contexts or to other carriers remains unverified. Moreover, while the feature set is rich, it does not include qualitative or manually captured information, such as ground staff availability, last-minute gate changes, or aircraft servicing delays, which may further impact turnaround time but are difficult to extract from structured records. Furthermore, the current modelling strategy also prioritised accuracy metrics such as RMSE, which guided the hyperparameter tuning process. While this yielded strong point predictions, it may have limited the calibration of the prediction intervals. Additionally, although the uncertainty estimates were generally well-behaved, the models exhibited a slight bias in coverage, with a tendency to overpredict. This suggests a possible benefit in explicitly including calibration objectives during training.

To address these limitations, future work should explore additional sources of context that may help stabilise performance during volatile periods. Calendar effects, such as the proximity to public holidays or major events, could serve as early indicators of staffing pressure or irregular passenger flows. Likewise, structured investigation into the most disrupted weeks in the test set could help clarify whether the residual variance is reducible, or instead reflects intrinsic limits to predictability in highly irregular operations. Next, improving calibration is another avenue for development. Rather than optimising solely for error reduction, future training strategies could incorporate loss functions that reward both accuracy and reliability. Targeting nominal coverage directly, or designing objective functions that penalise poorly calibrated intervals, may yield tighter and more trustworthy prediction bands, especially relevant when uncertainty estimates are to be used for operational decision-making. Lastly, while gradient boosting methods were chosen for their strong explainability and robustness in operational settings, there is scope to explore neural network-based approaches. Architectures that ingest sequential data, such as recurrent or convolutional networks, may uncover latent patterns in the temporal evolution of operational updates. These models could prove particularly valuable in capturing complex interactions or non-linear effects that current methods overlook.

9. Conclusion

Aircraft turnaround time remains one of the most volatile elements in airline operations and is a principal driver of reactionary delay, cost and excess emissions. This study was motivated by the need for a decision-support tool that can describe that volatility with both accuracy and clarity, offering controllers or operations optimiser algorithms not only a best estimate but also a credible indication of the residual risk.

The research has introduced a dynamic, probabilistic learning framework that fuses real-time airline, weather and air-traffic data into a single model. By focusing on one European hub carrier, this work exploited a rich and internally consistent feature set. The proposed model architectures deliver continuously updated conditional distributions of turnaround time, from several hours before gate arrival until the turnaround itself, and were deployed in an operational data pipeline to verify practical feasibility. Empirical testing shows a mean absolute error that falls from about 10 minutes immediately after the previous take-off to roughly 8 minutes as the aircraft reaches the stand, while the central 90% prediction interval contracts by more than a quarter across the whole prediction horizon. The model therefore answers the primary research question affirmatively: a probabilistic machine-learning framework can predict turnaround time distributions with useful precision and well-calibrated uncertainty. Analysis of feature contributions confirmed that available ground time, updated delay messages and congestion indicators are the dominant explanatory factors, clarifying the mechanisms identified in the supporting sub-questions.

These findings hold practical value for airline control centres. A continuously refreshed distribution exposes the most likely ground time and a confidence interval of the prediction, information that can inform gate allocation, slot requests and crew reassignment, or serve as direct input for optimisation engines such as a tail solver. Because the framework is generic in its handling of data streams, it could be extended to additional bases, long-haul rotations or collaborative airport settings. Further research might enrich the feature space with passenger flow or staffing variables, explore neural or sequence models that ingest temporal patterns explicitly, or refine interval calibration. A dedicated study of extreme outliers and of the interaction between predictive uncertainty and recovery strategies would also advance the contribution.

Appendix

Table 4. Overview of the features used in the turnaround time prediction model

Feature	Dynamic	Earliest information available
Numeric features		
Scheduled ground time in minutes	No	Before previous take off
Expected ground time	Yes	Before previous take off
Available ground time	Yes	Before previous take off
Minimum ground time for this turnaround	No	Before previous take off
Arrival time of day in minutes	No	Before previous take off
Passengers previous flight	Yes	Before previous take off
Passengers next flight	Yes	Before previous take off
Next flight load in kilogrammes	Yes	Before previous take off
Previous flight load in kilogrammes	Yes	Before previous take off
Expected arrival delay of previous flight	Yes	Before previous take off
Expected departure delay of previous flight	Yes	Before previous take off
Actual Off block delay previous flight	No	After previous take off
Actual On block delay previous flight	No	At turnaround start
Expected off block delay of next flight	Yes	After previous take off
Expected on block delay of next flight	Yes	After previous take off
CTOT minutes after departure next flight	Yes	Before previous take off
EC delay estimate previous flight	Yes	Before previous take off
Minutes until scheduled turnaround end	Yes	Before previous take off
Minutes until scheduled turnaround start	Yes	Before previous take off
Flight time previous flight	No	Before previous take off
Flight time next flight	No	Before previous take off
Number of inbound flights at airports (moving window)	No	Before previous take off
Number of outbound flights at airports (moving window)	No	Before previous take off
Wind speed	Yes	Before previous take off
Wind gusts	Yes	Before previous take off
Temperature	Yes	Before previous take off
Categorical features		
Previous departure airport	No	Before previous take off
Current destination airport	No	Before previous take off
Widebody aircraft indicator	No	Before previous take off
Aircraft subtype	No	Before previous take off
Day of week	No	Before previous take off
Month	No	Before previous take off
Departure terminal	No	Before previous take off
Visibility category	Yes	Before previous take off
Wind direction (cardinal)	Yes	Before previous take off
Freezing temperature indicator	Yes	Before previous take off
Rain or drizzle present	Yes	Before previous take off
Snow present	Yes	Before previous take off
Heavy weather phenomenon present	Yes	Before previous take off
Strong wind indicator	Yes	Before previous take off

Feature	Dynamic	Earliest information available
Strong easterly wind indicator	Yes	Before previous take off

References

[1] Ben Clatworthy. *European air traffic warning means summer delays for holidaymakers*. Accessed: 2025-07-06. 2025. URL: https://www.thetimes.co.uk/article/delay-warning-summer-holidays-air-traffic-control-gmj0bljbq.

[2] *Innovative Airline Operations: The Turnaround*. Accessed: 2025-01-27. 2023. URL: https://www.oag.com/blog/innovative-airline-operations-the-turnaround.

[3] EUROCONTROL. *EUROCONTROL Data Snapshot: Reactionary delays and their impact on airline operations*. Tech. rep. Accessed: 10-February-2025. EUROCONTROL, 2023. URL: https://www.eurocontrol.int/publication/reactionary-delays-data-snapshot.

[4] Eindhoven Airport. *Eindhoven Airport Uses AI to Improve Turnaround Process*. Accessed: 21-Feb-2025. Nov. 2023. URL: https://www.aviationpros.com/ground-handling/press-release/53077267/eindhoven-airport-eindhoven-airport-uses-ai-to-improve-turnaround-process.

[5] Passenger Terminal Today. *How is AI revolutionizing airports around the world?* Accessed: 21-Feb-2025. Jan. 2024. URL: https://www.passengerterminaltoday.com/features/exclusive-feature-how-is-ai-revolutionizing-airports-around-the-world.html.

[6] Mingchuan Luo, Michael Schultz, Hartmut Fricke, and Bruno Desart. "Data-driven Fusion of Turnaround Sub-processes to Predict Aircraft Ground Time". In: *Air Transport Research Society 2022 (ATRS)* (2022). URL: https://www.researchgate.net/publication/363153261_Data-driven_fusion_of_turnaround_sub-processes_to_predict_aircraft_ground_time.

[7] Yanyu Cui, Linyan Ma, Qingmiao Ding, Xuan He, Fanghui Xiao, and Bin Cheng. "Aircraft Turnaround Time Dynamic Prediction Based on Time Transition Petri Net". In: *PLOS ONE* 19.7 (2024), e0305237. DOI: 10.1371/journal.pone.0305237.

[8] Ehsan Asadi, Jan Evler, Henning Preis, and Hartmut Fricke. "Coping with Uncertainties in Predicting the Aircraft Turnaround Time at Airports". In: *Operations Research Proceedings 2019* (2020), pp. 773–780. DOI: 10.1007/978-3-030-48439-2_94.

[9] Paolino De Falco, Jan Kubat, Vladimir Kuran, José Rodriguez Varela, Salvatore Plutino, and Alessandro Leonardi. "Probabilistic Prediction of Aircraft Turnaround Time and Target Off-Block Time". In: *Proceedings of the 13th SESAR Innovation Days* (2023). URL: https://www.sesarju.eu/sites/default/files/documents/sid/2023/Papers/SIDs_2023_paper_26%20final.pdf.

[10] Michael Schmidt. "A Review of Aircraft Turnaround Operations and Simulations". In: *Journal of Air Transport Management* 63 (2017), pp. 34–40. DOI: 10.1016/j.jairtraman.2017.05.003.

[11] Cheng-Lung Wu and Robert E. Caves. "Modelling and Optimization of Aircraft Turnaround Time at an Airport". In: *Transportation Planning and Technology* 27.1 (2004), pp. 47–66. DOI: 10.1080/0308106042000184454.

[12] Xiaowei Tang, Jiaqi Wu, Cheng-Lung Wu, Ye Ding, and Shengrun Zhang. "Dynamic Prediction of Aircraft Turnaround Milestone Times Using a Cascaded Gradient Boosting Model for Improved Airport Collaborative Decision-Making". In: *Journal of Air Transport Management* 128 (2025), p. 102842. DOI: https://doi.org/10.1016/j.jairtraman.2025.102842.

[13] Hartmut Fricke and Michael Schultz. "Delay Impacts onto Turnaround Performance: Optimal Time Buffering for Minimizing Delay Propagation". In: *Proceedings of the 8th USA/Europe Air Traffic Management Research and Development Seminar (ATM 2009)* (2009). URL: https://www.researchgate.net/publication/262567633_Delay_Impacts_onto_Turnaround_Performance_-_Optimal_Time_Buffering_for_Minimizing_Delay_Propagation.

[14] Shervin AhmadBeygi, Amy Cohn, and Marcial Lapp. “Decreasing Airline Delay Propagation by Re-Allocating Scheduled Slack”. In: *IIE Transactions* 42.7 (2010), pp. 478–489. DOI: 10.1080/07408170903468605.

[15] Ramon Dalmau, Franck Ballerini, Herbert Naessens, Seddik Belkoura, and Sebastian Wangnick. “An explainable machine learning approach to improve take-off time predictions”. In: *Journal of Air Transport Management* 95 (2021), p. 102090. DOI: 10.1016/j.jairtraman.2021.102090.

[16] Ryota Mori. “Prediction of Off-Block Time Distribution for Departure Metering”. In: *Journal of Air Transportation* 32.3 (2024), pp. 122–129. DOI: https://doi.org/10.2514/1.D0359.

[17] Maarten Beltman, Marta Ribeiro, Jasper de Wilde, and Junzi Sun. “Dynamically forecasting airline departure delay probability distributions for individual flights using supervised learning”. In: *Journal of Air Transport Management* 126 (2025), p. 102788. DOI: 10.1016/j.jairtraman.2025.102788.

[18] Leo Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32. DOI: 10.1023/A:1010933404324.

[19] LightGBM Developers. *Python-package Introduction — LightGBM Documentation.* Accessed: 22 June 2025. 2025. URL: https://lightgbm.readthedocs.io/en/latest/Python-Intro.html.

[20] Tianqi Chen, Carlos Guestrin, and XGBoost Developers. *XGBoost Documentation, Version 3.0.2.* Accessed: 22 June 2025. 2025. URL: https://xgboost.readthedocs.io/en/stable/.

[21] Joannès Vermorel. *Pinball Loss Function Definition.* Accessed: 2025-06-22. Feb. 2012. URL: https://www.lokad.com/pinball-loss-function-definition/.

[22] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. “Optuna: A Next-Generation Hyperparameter Optimization Framework”. In: *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019, pp. 2623–2631.

[23] Scott M. Lundberg and Su-In Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems* 30 (2017). To appear in NeurIPS 2017. DOI: 10.48550/arXiv.1705.07874.

Part III

Operational Validation

*This part contains results that did not go into the scientific paper

7 Validation

Predictive performance claimed during development must hold once the model enters the operational environment. However, accuracy measured on historical snapshots often drifts when features are assembled in real time, because update lags, missing values and definition mismatches alter the joint distribution that reaches the predictor. The purpose of this chapter is therefore twofold: to quantify the gap between laboratory and production results and to identify the mechanisms that produce it.

Section 7.1 explains why conventional local experiments provide only a partial view and identifies sources of optimistic bias. Section 7.2 motivates the need for rigorous validation in the context of turnaround forecasting. Section 7.3 presents concrete failure modes observed after deployment, each of which eroded apparent accuracy despite favourable offline metrics. Section 7.4 then introduces an operational validation framework that recreates the live data pipeline, records point in time predictions and compares them with realised turnaround outcomes.

7.1. Why local results are incomplete

Local experiments generally rely on historical snapshots in which every feature is already available and perfectly aligned. During inference, however, many predictors must be reconstructed in real time through a sequence of database queries and joins. Even slight discrepancies in timestamps, versioning or filtering rules alter their values. Furthermore, the density of observations often increases as the moment of turnaround approaches, which introduces a sampling bias that favours short-horizon predictions in the test split. In practice the model must supply reliable estimates at earlier horizons as well, when information is sparse and uncertainty is larger. Without explicit validation at those horizons genuine performance remains unknown.

Figure 7.1 depicts the conventional route: a static dataset is divided into training and testing partitions, the model is fitted, and accuracy metrics are computed. The contrast with the operational inference workflow shown in Figure 7.2 highlights how delays, missing values and mismatches between feature definitions can erode performance once the predictor is embedded in the production pipeline.

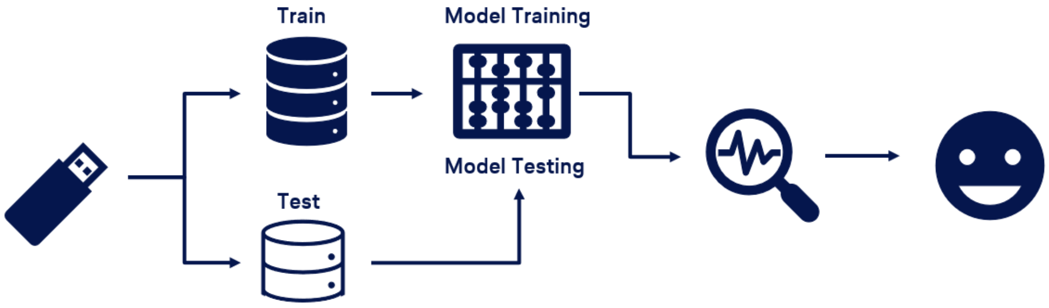


Figure 7.1: Conventional local training and testing workflow

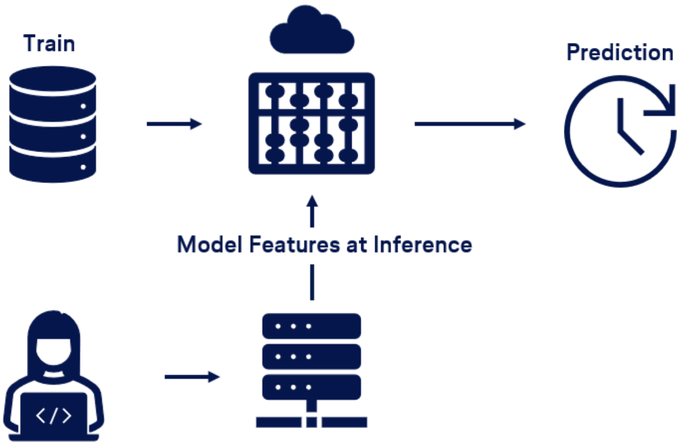


Figure 7.2: Operational inference workflow

7.2. Relevance for this thesis

A comprehensive validation strategy is indispensable because the research aims to supply decision makers with turnaround forecasts that remain trustworthy across the day-of-operation timeline. The wide variety of inputs increases the risk of hidden leakage, temporal drift and incomplete retrieval. Only by evaluating the model against the exact horizons and data pipelines used in production can credible bounds on error and uncertainty be established.

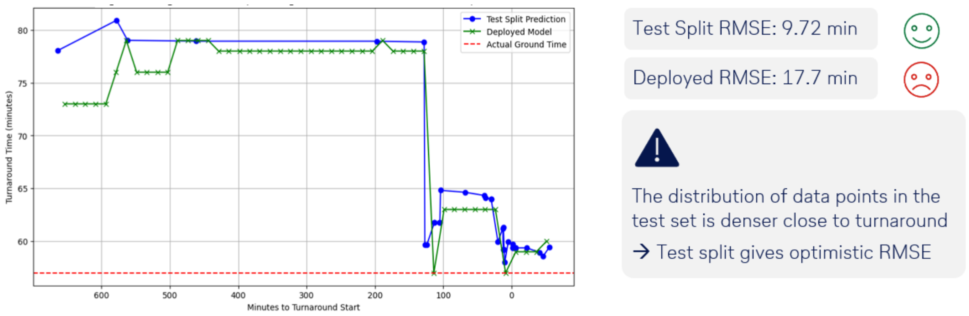
The subsequent sections cover what can possibly go wrong, and introduce a framework that reproduces the live environment, stores point-in-time predictions, and compares them with observed turnaround durations. Attention is devoted not only to the expected error but also to the calibration of the predictive distribution, since operational planners will rely on its lower bound when assessing feasibility. Adopting such a framework ensures that improvements reported in the scientific article translate directly into operational robustness rather than reflecting artefacts of an idealised laboratory setting.

7.3. Illustrative failure modes

Validation must probe the limits of the predictor rather than confirm its average success. Three concrete examples from the deployed system reveal shortcomings that would have remained hidden had evaluation relied solely on a conventional random hold-out split.

7.3.1. Horizon-dependent bias

Figure 7.3 contrasts predictions produced offline with those generated by the live service for the same flight. The root-mean-squared error (RMSE) on the test split appeared satisfactory, yet the error in production almost doubled because the test set contained proportionally more records collected shortly before turnaround. At that stage additional operational information (for example actual on-block time) reduces uncertainty, so the metric computed offline presented an overly optimistic picture. The lesson is clear: error must be reported at the specific horizons that matter to planners, not as a single aggregate across all lead times.



The distribution of data points in the test set is denser close to turnaround
→ Test split gives optimistic RMSE

Figure 7.3: Horizon-dependent bias. The offline test split contains a denser concentration of observations near the turnaround start, leading to an optimistic RMSE

7.3.2. Feature-retrieval inconsistency

Features are assembled at inference time through a sequence of joins across several operational databases, as depicted in Figure 7.4. Even minor discrepancies in timestamps, versioning, or naming conventions can alter their values. Figure 7.5 shows a case in which live predictions diverged sharply from those recorded during offline testing. Subsequent investigation revealed that the weather feed used in production lagged behind the archive employed during training by several minutes; the resulting mismatch propagated through engineered variables that describe recent precipitation intensity.

Such incidents underline the necessity of validating the full data-retrieval pipeline, not merely the statistical model. Automated checks now confirm that every feature served to the predictor in production matches the schema, type, and temporal alignment expected at training time.

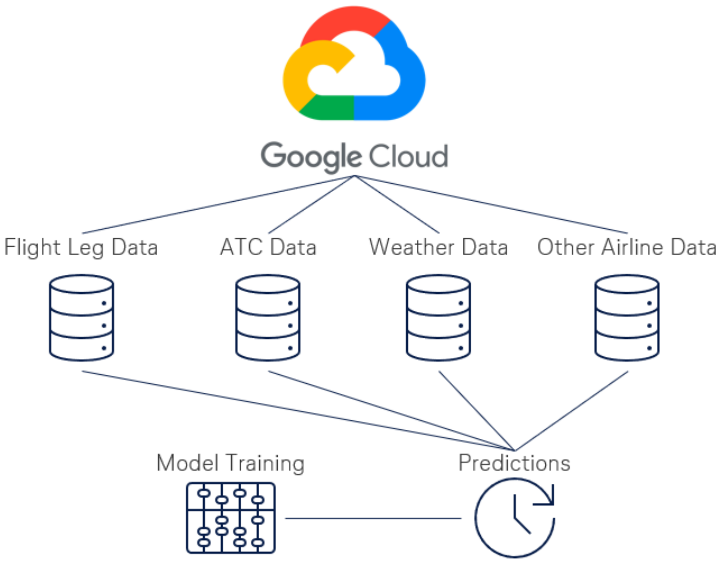


Figure 7.4: Data-retrieval workflow at inference. Flight leg, air-traffic-control, weather, and peer carrier data are dynamically joined before being supplied to the model.

7.3.3. Tail-swap anomaly

Occasional aircraft substitutions challenge any predictor that relies on pre-filed rotation plans. Figure 7.6 shows an episode in which the model initially forecast an excessively long turnaround because the inbound flight was matched to the onward leg of a different aircraft. A late tail-swap altered both destination and ground-handling sequence, prompting the deployed service to revise its estimate once the mismatch became evident. The offline test-split trajectory, assembled with post-event knowledge, contained the correct pairing from the start and therefore appeared reliable.

This incident reveals implicit label leakage in the archival dataset: target values were derived with

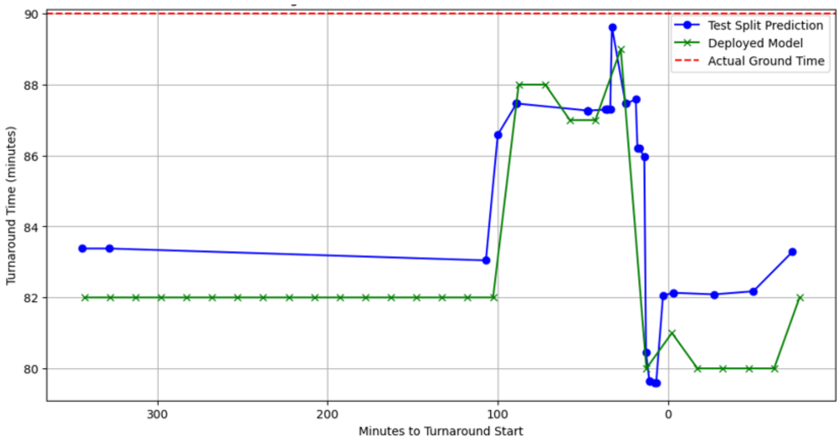


Figure 7.5: Feature-retrieval inconsistency. Live predictions (green) differ materially from offline estimates (blue) for the same flight because features were misaligned at inference.

hindsight of the true departure pairing, an advantage unavailable in live operation. Robust validation must replicate schedule uncertainty by withholding destination-specific attributes until confirmation is possible and by stress-testing historical data for similar tail-swaps.

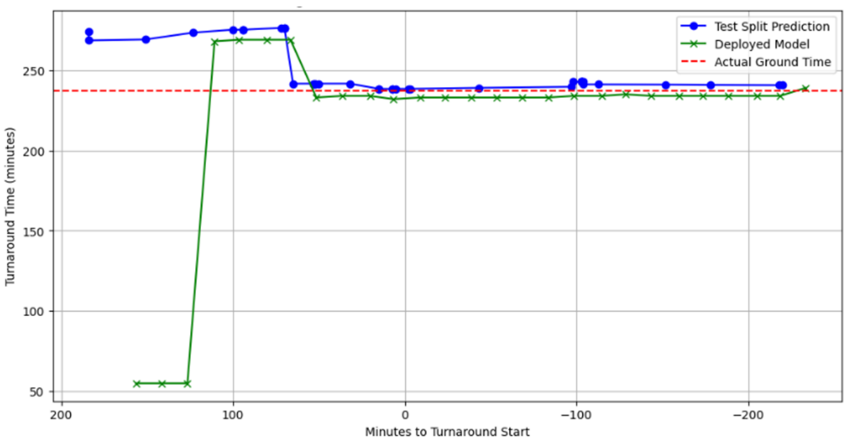


Figure 7.6: Tail-swap anomaly. The deployed predictor (green) corrects its estimate once the substitution becomes apparent, whereas the offline test-split prediction (blue) benefits from retrospective knowledge of the actual departure pairing.

7.4. Operational Validation Framework and Results

This section summarises how live forecasts were captured, matched to realised turnarounds, and benchmarked against an offline LightGBM run. Using three weeks of operations marked by irregularities, the performance gap is quantified.

7.4.1. Operational validation framework

Operational validation replicates live decision making by capturing every forecast that the deployed service produced during day of operations and comparing it with the turnaround that later materialised. The service runs as a scheduled job inside the airline's platform. For each inbound flight it starts issuing forecasts 24 hours before the advertised on block time and refreshes them at fixed 15 minute intervals until the aircraft leaves the stand. Each call assembles the latest schedule, weather, traffic, and ground process data, constructs the feature vector, and writes the resulting distribution together with a precise timestamp, prediction horizon, and flight identifier to the validation store.

Once the aircraft has departed, the realised turnaround time is derived from the recorded on block and off block times in the flight movement database. This value is joined with the corresponding time stamped predictions, allowing direct assessment of accuracy, bias, and calibration at every horizon that matters to the operations control centre.

To permit a fair comparison with the results previously reported for the local benchmark model in the scientific article, the identical set of turnarounds is retrieved from the archives and processed through the offline evaluation script. Both workflows therefore encompass the same flights and horizons; the only difference lies in the data retrieval pathway. The analysis window spans 15 June 2025 to 8 July 2025, the period for which complete deployed forecasts are available. This local test set differs from that used in the scientific article and relates to a period marked by frequent operational irregularities, which produces weaker performance than reported earlier in the scientific article.

7.4.2. Deployed versus local error

Figure 7.7 and Figure 7.8 contrast mean absolute error and root mean squared error recorded by the service in live operation with those achieved by the same LightGBM model when run locally on the identical flights. Two perspectives are shown: lead time measured from the forthcoming turnaround start, and lead time measured from the previous off-block minute.

Across the full horizon the deployed service under-performs the offline benchmark. The gap between the test split and deployed is around 5 - 7 minutes in MAE and exceeds 25 minutes in RMSE, evidence that misaligned or stale features expose the predictor to distribution shift. Even within the last hour before pushback the live curve remains significantly above the local one.

This pattern corroborates the failure modes outlined in Section 7.3. Feature-retrieval latency and schedule uncertainty all can lead to higher operational error. However, as the local training and test data have extensively been reviewed for data leakage, it is suspected that there is an undefined error with feature retrieval at inference. Therefore, Improvement efforts must target tighter alignment between training and inference features, resilience to schedule volatility and early-horizon calibration so that the deployed error curve approaches the reference without compromising real-time constraints.

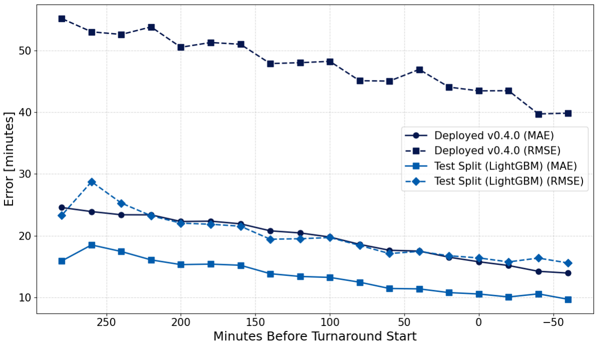


Figure 7.7: Error over prediction horizon measured from turnaround start.

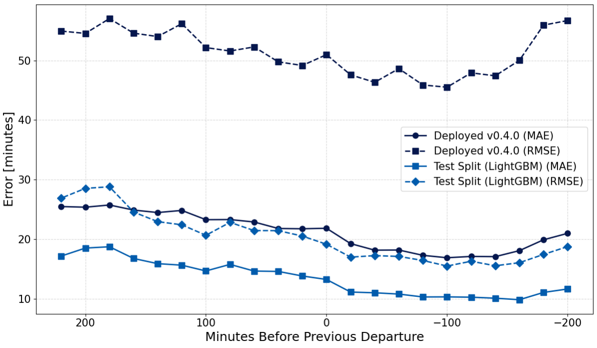


Figure 7.8: Error over prediction horizon measured from previous departure.

7.4.3. Deployed versus local uncertainty

Figure 7.9 and Figure 7.10 compare the average width of the ninety per cent prediction interval, defined as the difference between the 5th and 95th quantiles. Horizons are again expressed relative to the forthcoming turnaround start and to the previous departure.

The deployed service produces wider intervals across most lead times. Although the gap seldom exceeds four minutes, the pattern is systematic: the live curve lies above the local benchmark for nearly every point on both scales. The discrepancy implies that the deployed model expresses greater uncertainty, which suggests that crucial contextual information is either delayed or absent at inference time.

Greater interval width is not necessarily undesirable, planners prefer a truthful expression of uncertainty over an overly narrow one, but the divergence indicates that the calibration applied during training may not propagate to production unless the feature pipeline is harmonised. Efforts aimed at reducing systematic feature loss or delay are expected not only to lower the error curves in Figure 7.7 and Figure 7.8 but also to bring the operational interval width in line with the local reference.

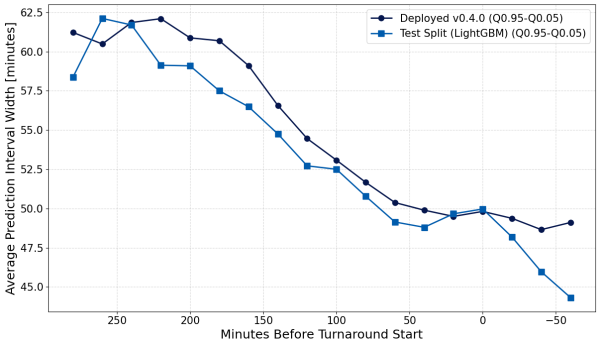


Figure 7.9: Average width of the ninety□per□cent prediction interval as a function of lead time before turnaround start.

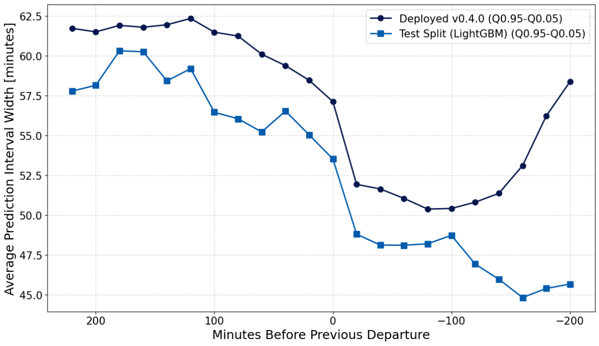


Figure 7.10: Average width of the ninety□per□cent prediction interval as a function of lead time before previous departure.

Part IV

Closure

8 Conclusions & Recommendations

8.1. Conclusions

The research questions posed in Chapter 4 are repeated below for convenience.

Research Question 1

How accurately can a probabilistic machine learning framework dynamically predict aircraft turnaround time distributions?

To address the main research question (RQ 1), the study explored the following sub-questions:

RQ1.1 What are the key operational and environmental factors affecting aircraft turnaround time?

Across the entire prediction horizon the most influential variable proved to be the *available ground time*. Once the inbound sector had departed, dynamic delay indicators rose in importance: the actual off-block delay of the previous flight, the estimate of its expected arrival delay, and the Calculated Take-Off-Time offset for the outbound leg all ranked immediately after available ground time in Shapley attribution. Passenger load and total freight mass seemed to gain importance during the final hour before stand arrival, reflecting the workload imposed onboarding and loading activities. Structural descriptors such as aircraft subtype, wide-body flag, departure and arrival airports, and coarse congestion metrics retained secondary relevance; weather variables affected the extreme tails only on the rare occasions of strong wind, snow, or heavy rain. Together these findings confirm that turnaround duration is governed by a hierarchy in which schedule design sets the baseline, real-time delay propagation dictates the mid-horizon adjustments, and load-related factors dominate the last tactical refinements.

RQ1.2 Which probabilistic modelling techniques are best suited to capture and quantify the inherent uncertainty in turnaround operations?

Among the algorithms assessed, gradient-boosted decision-tree ensembles trained with quantile regression delivered the best results, outperforming Random Forest on error metrics. LightGBM attained a median absolute error of 10 minutes immediately after the preceding take-off, falling to 8 minutes at on-block, while its central 90% interval remained within 2 percentage points of nominal coverage. XGBoost showed comparable point accuracy but produced wider and less responsive prediction intervals. Split models, trained on long- versus short-haul rotations or on slack versus non-slack schedules, narrowed the predictive interval by three minutes on average but yielded no meaningful reduction in point error, indicating limited incremental benefit over a single well-tuned LightGBM.

RQ1.3 What validation strategies and performance metrics can be employed to assess the framework's effectiveness using operational data?

Effectiveness was first evaluated offline on a chronologically separated test set: records from January 2024 to March 2025 formed the training sample, while turnarounds between April and June 2025 served as hold-out data. Mean absolute error, root-mean-square error, interval width and empirical coverage were computed over the full horizon and at two operationally salient instants, namely the moment the inbound flight left the gate and the start of the ground phase. Bucketed time-to-event profiles traced performance evolution every twenty minutes, ensuring that the non-uniform density of updates did not obscure horizon-dependent effects.

Beyond this local testing, the broader thesis project incorporated deploying the model. The trained model was embedded in the airline's data pipeline, where it schedules predictions for all hub turnarounds up to twenty-four hours before arrival and refreshes them every fifteen minutes. Real-time SQL queries assemble features from several operational systems, and the resulting forecasts, together with realised outcomes, are persisted for continuous monitoring of accuracy, coverage, calibration and drift. This live set-up reproduces inference-time challenges such as data latency, schema evolution and occasional upstream failure, and flags any degradation relative to the offline benchmarks.

RQ1.4 How does the predictive uncertainty evolve between the moment of take-off of the previous flight in the turnaround, and its landing?

Predictive uncertainty, expressed as the width of the 5th to 95th percentile band, contracted markedly as the horizon shortened. For LightGBM the average band narrowed by 28% between the preceding departure and on-block, with over half of that reduction occurring within the first 30 minutes after take-off, once the actual off-block time of the inbound leg became known. Thereafter, the interval decayed gradually while the aircraft was airborne, and levelled off during the turnaround itself because most high-impact features were already resolved. Empirical coverage held steady around eighty-nine per cent throughout, indicating that the shrinking band reflected genuine information gain rather than overconfidence. A mild skew persisted: the lower half of the distribution contained slightly more realisations than the upper half, showing a systematic tendency to over-predict when uncertainty was high, but this bias diminished as richer operational data arrived. Consequently, the framework furnished controllers with increasingly tight and well-calibrated distributions, enabling proportionate mitigation actions at progressively lower risk.

Taken together, these answers demonstrate that a probabilistic machine-learning framework based on gradient-boosted quantile regression can predict aircraft turnaround time distributions with operationally useful accuracy and reliability, while providing transparent insight into the factors that matter most. Evaluation on past turnarounds yields a median absolute error of 10 minutes immediately after the preceding take-off, falling to 8 minutes at on-block. Results show that the uncertainty of the prediction reduces by a quarter as updated operational data like delays or air-traffic control slots come in.

8.2. Recommendations

Building on the limitations and future directions identified earlier in the scientific article, several improvements can be made. First, the model could benefit from a broader set of features to better reflect the operational context. For example, adding indicators for public holidays or major events could help the model handle unexpected schedule changes and maintain more stable performance during busy periods. A useful follow-up study would be to examine the most disrupted weeks in the test set in more detail. This could help determine whether the remaining prediction error is due to missing features or simply reflects the unpredictable nature of certain operational conditions.

Second, the dataset could be extended to include outstation turnarounds and, where possible, data from other airlines. This would help assess how well the model generalises beyond the current hub context. Including more qualitative information, such as staff availability, last-minute gate changes, or servicing delays, could also improve the accuracy of predictions.

Future modelling efforts should also focus on both accuracy and uncertainty. Rather than only optimising for error metrics like RMSE, it would be valuable to test hyperparameter tuning that encourage well-calibrated prediction intervals. This could make the uncertainty estimates more reliable and more useful for decision-making. Given the time-based nature of the data, it may also be worthwhile to explore neural network architectures, such as recurrent or convolutional models, that are better suited to capturing time-dependent patterns.

Finally, as shown in the validation results, ensuring consistent model performance after deployment is crucial. This requires close monitoring of the data pipelines to make sure that the features used during live predictions match what the model expects. By addressing these areas, improving the feature set, extending the dataset, enhancing calibration, exploring alternative models, and ensuring robust deployment, the framework can become an even more reliable and operationally useful tool.