Ca' Foscari
University
of Venice

Master's Degree

in Data Analytics for Business and Society

Final Thesis

The Field Technician Scheduling Problem:

A Predict-then-Optimise Approach Using Machine Learning and Constraint Programming

Supervisor
Ch. Prof. Raffaele Pesenti

Graduand
Subodh Khadka
Matriculation Number 900945

Academic Year
2025 / 2026

Acknowledgements

I would like to sincerely thank my supervisor, Professor Raffaele Pesenti, for his invaluable guidance, patience, and constructive feedback throughout the development of this thesis. I am also grateful to Ca' Foscari University of Venice for providing an environment that fostered my academic and professional growth.

My deepest gratitude goes to my mother. Her endless sacrifices and unwavering belief in me made this milestone possible, and I dedicate this achievement to her.

Finally, I am incredibly thankful for my family's constant support, which kept me grounded and focused every step of the way.

Contents

Acknowledgements iii

Abstract vii

1 Introduction 1
1.1 Industry Context and Field Service Management . . . 1
1.2 The Scheduling Problem in Operations Research . . . 1
1.3 Two Planning Assumptions . . . 2
1.4 The Cross-Border Industrial Maintenance Scenario . . . 3
1.5 Research Questions and Objectives . . . 3
1.6 Contributions of the Thesis . . . 4
1.7 Use of Artificial Intelligence Tools . . . 5
1.8 Thesis Structure . . . 5

2 Literature Review 7
2.1 From Vehicle Routing to Field Technician Scheduling . . . 8
2.2 Solution Methods and Horizon Decomposition . . . 9
2.3 Machine Learning for Service-Duration Prediction . . . 10
2.4 The Predict-Then-Optimise Paradigm . . . 11
2.5 Decision Support and Digital-Twin Concepts in Field Service Management . . . 13
2.6 Research Positioning and Scope . . . 14

3 Methodology and System Architecture 15
3.1 Synthetic Data and Scenario Design . . . 15
3.1.1 Rationale for Synthetic Data . . . 16
3.1.2 Input Datasets and Their Relationships . . . 16
3.1.3 Geographic Representation and Dataset Scale . . . 18
3.2 System Architecture and Prototype Design . . . 19
3.2.1 Overall System Architecture . . . 19
3.2.2 Technology Stack and Design Choices . . . 19
3.2.3 Prototype Development Process . . . 20
3.2.4 User Interface and Decision-Support Functions . . . 21
3.3 Predictive Layer: Service-Duration Estimation . . . 24
3.3.1 Prediction Problem and Target Variable . . . 24
3.3.2 Feature Engineering . . . 25
3.3.3 Model Selection . . . 26
3.3.4 Validation Strategy . . . 26
3.3.5 Predictive Uncertainty and Safety Buffers . . . 27
3.4 Prescriptive Layer: CP-SAT Technician Scheduling . . . 27
3.4.1 Scheduling Problem and Mathematical Formulation . . . 28
3.4.2 Constraints and Objective Function . . . 29

3.4.3 Skill Compatibility and Capacity Feasibility . . . 29
3.4.4 Daily Decomposition and Multi-Day Reservations . . . 31
3.4.5 Route-Display Post-Processing . . . 31
3.5 Risk Scoring and Experimental Design . . . 31
3.5.1 Composite Risk Scoring . . . 32
3.5.2 Duration Regimes and Ablation Design . . . 32
3.5.3 What-If Scenario Simulation . . . 33
3.6 Verification and Evaluation Scope . . . 34

4 Operational Diagnosis and Planning Implications 35
4.1 Geographic Distribution and Regional Capacity Imbalances . . . 35
4.2 Duration Bias and Prediction Errors . . . 38
4.3 Historical SLA Lateness and Associated Factors . . . 40
4.4 Travel Requirements and Their Planning Implications . . . 42
4.5 Job Demand and Skill Bottlenecks . . . 43
4.6 Capacity Bottlenecks and the Four-Lever Response . . . 45
4.7 From Diagnosis to Planning Decisions . . . 46

5 Results and Discussion 49
5.1 Duration-Prediction Results . . . 49
5.1.1 Model Comparison and Prediction Errors . . . 49
5.1.2 Comparison with Calibrated Static Estimates . . . 51
5.2 Scheduling Results under Alternative Duration Estimates . . . 52
5.2.1 Interpretation of Planning Indicators . . . 52
5.2.2 Static and ML-Based Scheduling . . . 53
5.2.3 Calibrated Estimates and Safety Buffers . . . 55
5.2.4 Solver Performance and Computational Time . . . 56
5.3 Response to Operational Disruptions . . . 56
5.3.1 Single-Factor Sensitivity . . . 56
5.3.2 Combined Disruption Scenarios . . . 58
5.4 Discussion and Planning Implications . . . 59

6 Conclusions, Limitations and Future Work 61
6.1 Conclusions by Research Question . . . 61
6.2 Limitations . . . 62
6.2.1 Data and Predictive Validation . . . 62
6.2.2 Scheduling Representation and Planning Indicators . . . 63
6.2.3 Scenario Comparisons and Cost Accounting . . . 64
6.2.4 Prototype Validation and Reproducibility . . . 64
6.3 Future Work . . . 65

List of Abbreviations 67

List of Figures 68

List of Screenshots 69

List of Tables 70

References 71

Abstract

The Field Technician Scheduling Problem (FTSP) in Field Service Management (FSM) requires assigning skilled technicians to geographically dispersed tasks under strict Service Level Agreements (SLAs) and finite capacity. This thesis investigates the predict-then-optimise paradigm by developing a decision support prototype that bridges data-driven duration estimation and exact Constraint Programming. A Random Forest model, trained on 3,000 synthetic historical assignments, predicts job durations with a cross-validated coefficient of determination (R^2) of 0.748, correcting a systematic duration bias: actual historical durations exceed planned durations by 14.8% on average. Benchmarking shows that most of this gain comes from removing that bias rather than from model complexity: linear regression scores marginally better on aggregate error, and the tree-based ensemble is retained as the implemented predictor, with feature-importance summaries providing operational inspection. These predicted durations drive a Google OR-Tools CP-SAT model that schedules technicians subject to skill compatibility and daily capacity limits, with multi-day reservations and penalties for unfilled crew slots. The integration yields a critical diagnostic result: under a formulation requiring each job to fit within a single technician-day, the modelled workload renders the schedule largely infeasible, with 762 jobs exceeding the daily limit. A four-lever capacity strategy incorporating multi-day rolling carryover, selective overtime, targeted workforce augmentation, and priority-weighted penalties recovers coverage to 851 jobs under ML inputs and 904 jobs under static inputs, with jobs receiving no technician assignment confined to normal priority. To maintain computational tractability, the 30-day horizon is decomposed into sequential daily scheduling windows rather than solved as a monolithic global optimisation, and travel is modelled as a fixed cost from each technician base to the job site without optimising inter-job routes. Within this bounded scope, the framework enables interactive what-if scenario analysis, allowing management to stress-test capacity assumptions and shift from reactive dispatching toward proactive workforce planning.

Chapter 1

Introduction

1.1 Industry Context and Field Service Management

Field Service Management (FSM) coordinates the scheduling, dispatching, and execution of technical interventions at customer locations. The work spans diagnostic visits, corrective repairs, scheduled maintenance, equipment installations, and compliance inspections across industries that depend on asset uptime: telecommunications, energy, manufacturing, healthcare, transportation, and logistics [1], [2]. Every working day, a finite pool of technicians must be matched to geographically dispersed service requests under hard constraints on skills, contractual deadlines, travel feasibility, and labour cost [3].

Scheduling decisions connect service delivery with labour availability, travel expenditure and contractual expectations. Workforce scheduling and routing studies show that these decisions must be considered together when skills and geography restrict the assignments available to a planner [1], [3]. Commercial market reports also describe FSM as a growing software category [4], [5]; these forecasts place the research within the wider development of service-management software.

A dispatcher needs to understand not only which assignments are selected, but also why some demand remains unallocated and how a change in the workforce or planning assumptions changes the result. This thesis investigates that decision-support use case through a controlled synthetic scenario, focusing on the relationship between duration estimates, capacity assumptions and generated assignments.

1.2 The Scheduling Problem in Operations Research

From an Operations Research (OR) perspective, field service planning that jointly determines technician assignments and travel sequences is related to the *Vehicle Routing Problem* (VRP) family [2]. The VRP with Time Windows extends the base model with service intervals, closely mirroring Service Level Agreement (SLA)-driven field operations [2]. The *Technician Routing and Scheduling Problem* (TRSP) adds three further properties: skill

compatibility (only a subset of the workforce qualifies for each task [3], [6]), performance heterogeneity (experienced technicians complete identical tasks faster [6]), and side constraints including spare-parts availability, two-technician synchronisation, and multi-day interventions [1], [3], [6].

The problem addressed in this thesis is a *Field Technician Scheduling Problem* (FTSP): technicians are assigned to jobs subject to skill, capacity, multi-day reservation, and crew requirements. Travel enters the model through base-to-job time and cost estimates. Job-to-job route sequences are not decision variables, so the implementation does not solve the full TRSP. The routing literature provides related context for the scheduling formulation.

Exact, heuristic and hybrid methods address different technician-routing formulations. Pillac et al. [7] combine Adaptive Large Neighbourhood Search with mathematical-programming post-optimisation; broader metaheuristic approaches are reviewed by Toaza and Esztergár-Kiss [8]. Recent work reframes TRSP as a decision-support problem, emphasising what-if analysis of workforce sizing and skill upgrades [9]. The pipeline adopted here first predicts durations using Machine Learning (ML) and then constructs assignments using the CP-SAT constraint-programming solver. This follows the *predict-then-optimise* paradigm [10], [11].

1.3 Two Planning Assumptions

Before introducing the scheduling model, two initial planning assumptions need to be defined. They simplify the general scheduling task and, as the results later show, they have an important effect on the number of jobs that can be allocated. The first concerns how job duration is estimated, while the second concerns whether a job can be completed within one working shift.

Static duration estimates. Each job is initially assigned an expected duration that is used as a fixed planning value. However, in the synthetic historical data, these estimates tend to underestimate the actual time needed to complete an intervention. This can lead the scheduler to plan more work than technicians can realistically perform. For this reason, the thesis compares the original static estimate with two alternatives: a calibrated multiplier and a Machine Learning prediction. Service duration may also vary according to the technician assigned, especially when technicians have different levels of experience [6].

A single-shift feasibility rule. In the initial formulation, a job can only be assigned if its service time and travel requirement can fit within one technician's working shift. Therefore, jobs requiring more time than a single shift are treated as infeasible under this rule. This does not mean that such jobs cannot be completed in real operations; it only reflects a limitation of the initial scheduling formulation. Later in the thesis, multi-day reservations are introduced to relax this restriction. These reservations should still not be interpreted as a detailed execution schedule, since the model does not explicitly optimise start times,

breaks, or the synchronisation of two-technician crews.

1.4 The Cross-Border Industrial Maintenance Scenario

This thesis examines a synthetic but operationally plausible scenario involving a mid-sized European industrial field service provider. The provider serves manufacturing, logistics, and data centre clients across Italy, Germany, France, Spain, and the United Kingdom. The workforce initially consists of 50 technicians distributed across the five countries. Each technician holds skills from an eight-competency vocabulary: pumps, motors, programmable logic controllers (PLCs), robotics, sensors, networks, hydraulics, and drives. The job pool contains 1,000 scheduled interventions over a 30-day planning horizon. These range from routine preventive inspections to critical breakdown responses, with SLA windows varying from a few hours to several days.

The scenario focuses on cross-border industrial maintenance rather than local routing. It is designed to examine what happens when technicians, specialised skills, and customer demand are geographically dispersed across Europe. In this type of environment, a company that can allocate the right technician to the right job more effectively can make better use of limited specialist capacity and improve its responsiveness to customers. This can become an important operational advantage compared with less coordinated competitors. In the dataset, 85% of all technician–job pairings exceed the 800 km flight threshold, while the mean travel distance is 1,140 km. These figures refer to all possible technician–job pairings, not only to the journeys selected in the final schedule. In practice, the nearest skill-compatible technician is often much closer than the average candidate.

The primary user of the prototype is a dispatcher. The dispatcher can select a planning horizon and duration regime, generate assignments, inspect the map and schedule board, and compare scenario summaries. Managers can use the same outputs to identify capacity and skill shortages. Machine attributes are included to describe the service jobs, but the prototype is not intended to be a complete asset-management or failure-prediction system.

1.5 Research Questions and Objectives

The main objective of this thesis is to investigate whether data analytics techniques, in particular Machine Learning (ML) and Constraint Programming (CP), can improve the planning of field service operations compared with schedules based on fixed duration estimates. Rather than replacing the judgement of dispatchers or operational experts, the proposed system is intended to support their decisions by combining data-driven duration estimates with optimisation-based scheduling.

The scheduling environment is implemented as an interactive decision-support prototype. It allows users to generate schedules under different duration assumptions, inspect

the resulting assignments, and compare operational indicators such as assignment coverage, travel time, reported cost, and SLA-related measures. The prototype also supports the diagnosis of capacity constraints and the testing of operational scenarios, helping managers understand where the main scheduling pressures occur.

Four research questions guide the study:

RQ1: How accurately can a Machine Learning model predict actual job durations compared with fixed FSM duration estimates?

RQ2: How do alternative duration estimates affect assignment coverage, reported cost, travel time, and the SLA and utilisation indicators produced by the scheduling model?

RQ3: What structural capacity bottlenecks does the scheduling model reveal, and to what extent can targeted operational adjustments improve assignment coverage?

RQ4: How does the scheduling system respond to changes in workforce availability, travel time, spare-parts availability, and daily shift capacity?

To address these questions, the thesis has four main objectives:

1. To evaluate whether ML-based duration estimates improve on fixed planning estimates.
2. To examine how different duration estimates change the schedules produced by the CP-SAT model.
3. To identify the main capacity and skill bottlenecks in the synthetic field service scenario and test possible operational adjustments.
4. To evaluate the system under simulated disruptions affecting workforce availability, travel conditions, spare parts, and daily capacity.

1.6 Contributions of the Thesis

This thesis makes three main contributions.

1. It develops a predict-then-optimise approach built entirely on open-source tools for field technician scheduling. The approach combines a Random Forest model for estimating job duration with a CP-SAT model for assigning technicians to jobs. It shows how data-driven duration estimates can be incorporated into scheduling without relying on commercial solvers or proprietary company data.

2. It uses the scheduling model as a diagnostic tool. The initial results show that the single-shift feasibility rule creates a major capacity restriction in the scenario. The model is then used to test practical adjustments, including additional technicians, overtime, multi-day reservations, and priority-based penalties.

3. It implements the approach in an interactive web-based decision-support prototype. The prototype allows dispatchers and managers to generate schedules, inspect assignments, compare duration assumptions, and test operational scenarios. It can therefore be used internally as a planning and diagnostic tool for field service operations.

1.7 Use of Artificial Intelligence Tools

Artificial intelligence tools supported the development of the prototype and the writing of this thesis. NotebookLM and Google Gemini were used to organise working notes, clarify technical concepts, and explore ideas. ChatGPT was used to brainstorm design alternatives, discuss the research framework, and help express the author's ideas more clearly. During writing, this included assistance with drafting, paraphrasing, and revising passages across the thesis. Claude Code supported the implementation and debugging of the prototype described in the methodology chapter.

These tools helped the author explore possible approaches and refine the work. The final decisions on the research questions, objectives, modelling choices, experimental design, and interpretation of results remained with the author. The literature search, selection and reading of academic papers were carried out by the author, who also checked the citations and the claims supported by those sources.

AI-generated suggestions, text, and code were reviewed and adapted before inclusion. Written content was checked for consistency with the intended meaning, the cited sources, and the reported results. Suggested code was reviewed in relation to the intended behaviour of the prototype and revised where necessary. AI outputs were treated as support material requiring verification, rather than as independent evidence. The author takes responsibility for the final content, analysis, and conclusions.

1.8 Thesis Structure

This thesis is organised as follows. Chapter 2 reviews the literature on workforce scheduling, technician routing, duration prediction, and decision support. Chapter 3 describes the synthetic data, system implementation, Machine Learning model, and CP-SAT scheduling formulation. Chapter 4 presents the capacity, geographic, travel, and skill diagnostics used to identify the main operational bottlenecks. Chapter 5 reports the predictive evaluation, scheduling results, and what-if scenario experiments. Finally, Chapter 6 summarises the

main findings, discusses the limitations of the prototype, and outlines directions for future work.

Chapter 2

Literature Review

This chapter presents a targeted review of the literature relevant to field technician scheduling. The review was conducted using Google Scholar and the Ca' Foscari University Library System's *Banche dati* electronic resources. Searches combined terms related to the application domain, including "Field Service Management", "field technician scheduling", "Technician Routing and Scheduling Problem", and "workforce scheduling", with methodological terms such as "Machine Learning", "service duration prediction", "Constraint Programming", "CP-SAT", "decision support systems", and "digital twin".

Priority was given to recent peer-reviewed journal articles and conference papers that directly relate to the problem addressed in this thesis. Earlier studies were retained when they provide the main theoretical or methodological foundations for technician scheduling, routing, optimisation, or predictive modelling. The aim of the review is not to cover every aspect of field service operations, but to identify the techniques used in previous studies and explain the choices made in the proposed prototype.

The main focus of this thesis is the Field Technician Scheduling Problem (FTSP): assigning the right technician to the right job while considering skills, available capacity, estimated service duration, and service requirements. Routing literature is included to position the problem within the wider Operations Research literature, since travel affects the feasibility and cost of an assignment. However, the prototype does not solve a complete routing problem with optimised job-to-job sequences. Instead, it focuses on technician assignment and capacity planning using base-to-job travel estimates.

The remainder of the chapter is organised around the main elements of the proposed approach. First, it positions technician scheduling in relation to vehicle-routing and workforce-scheduling problems. It then reviews exact, heuristic, and decomposed optimisation approaches. The following sections discuss Machine Learning methods for duration prediction, the predict-then-optimise paradigm, and decision-support and digital-twin concepts. The chapter concludes by defining the research position and scope of the thesis.

2.1 From Vehicle Routing to Field Technician Scheduling

Field service scheduling is related to the Vehicle Routing Problem (VRP), but it involves more than deciding which vehicle should visit which customer. In a traditional VRP, the main concern is usually the sequence of visits, vehicle capacity, and travel distance. In field service operations, however, each intervention may require a technician with a specific technical skill, level of experience, or availability. The scheduling problem is therefore not only about reaching a customer location, but also about assigning the right technician to the right job.

The VRP literature provides the starting point for this type of problem. The Vehicle Routing Problem with Time Windows (VRPTW) adds the requirement that customers must be served within specific time intervals, while the Multi-Depot VRP with Time Windows considers vehicles or workers departing from different locations. These features are relevant to field service because technicians may start from different bases and jobs may be subject to Service Level Agreement (SLA) deadlines. Konstantakopoulos et al. [2] provide a recent classification of VRP variants and solution methods, showing how practical requirements such as time, capacity, and geography increase the complexity of routing decisions.

The literature gradually moved from general routing problems towards models that recognise the specific characteristics of the workforce. An early and important example is the study by Cordeau et al. [3], which considers the scheduling of technicians and tasks in a telecommunications company. Their work shows that technicians cannot be treated as interchangeable resources, because task requirements and technician qualifications limit the set of feasible assignments. Pillac et al. [7] further develop this idea through the Technician Routing and Scheduling Problem (TRSP), where skills, tools, spare parts, time windows, and route feasibility must be considered together. These studies are important because they show how service operations combine geographical constraints with workforce constraints.

Castillo-Salazar et al. [1] provide a broader review of workforce scheduling and routing problems. Their study highlights that workforce scheduling differs from standard vehicle routing because workers have qualifications, working-hour restrictions, and service requirements that directly affect feasibility. This distinction is particularly relevant to the present thesis. A technician may be geographically close to a job, but still be unable to perform it because the required skill is missing or because the technician does not have enough remaining capacity during the working day.

More recent research has also considered differences in technician performance. Yu et al. [6] study the Field Technician Scheduling Problem with experience-dependent service times. Their model recognises that the duration of the same intervention may vary depending on the technician who performs it. This is the study most closely related to the modelling choices in this thesis, since it supports the idea that service duration should not

always be treated as a fixed characteristic of the job. Instead, technician experience and job characteristics can influence the time required and, consequently, the capacity available for other assignments.

The literature reviewed in this section therefore supports three main choices in the proposed model: skill compatibility between technicians and jobs, daily capacity restrictions, and duration estimates that can vary across assignments. Travel is also included because it affects the time and cost of serving a job. However, the present thesis does not solve a complete TRSP or optimise job-to-job route sequences. It focuses on the Field Technician Scheduling Problem (FTSP), using base-to-job travel estimates to support daily technician assignment and capacity planning. Detailed routing and visit sequencing remain relevant extensions for future work.

2.2 Solution Methods and Horizon Decomposition

Scheduling problems can be solved using different approaches, depending on the level of detail required and the size of the instance. The main distinction is between exact methods, which can prove that a solution is optimal for a given formulation, and heuristic or metaheuristic methods, which aim to find good solutions within a limited computation time. In technician scheduling, the choice depends on the number of jobs and technicians, the routing detail included, and the operational constraints that must be represented.

Exact optimisation methods include Mixed-Integer Programming (MIP) and Constraint Programming. These approaches are useful when the model contains hard restrictions that cannot be ignored, such as skill compatibility, daily working capacity, crew requirements, or time limits. Planning and scheduling research shows that exact methods can provide strong guarantees, but their computational requirements increase rapidly as the number of decisions and constraints grows [12]. In this thesis, Google OR-Tools CP-SAT is used to model binary technician–job assignments and integer capacity constraints. CP-SAT is well suited to this formulation because it can search for feasible schedules while enforcing the operational restrictions directly [13].

When routing and scheduling decisions become more detailed, exact approaches may become difficult to apply to large instances. For this reason, many studies use heuristics and metaheuristics. Pillac et al. [7] propose a parallel matheuristic for the Technician Routing and Scheduling Problem, combining large-neighbourhood search with mathematical-programming post-optimisation. Their work shows how a heuristic search can explore many possible routing and assignment combinations, while the mathematical-programming stage improves promising solutions. More recently, Toaza and Esztergár-Kiss [8] review 120 metaheuristic applications to TSP-based scheduling problems, including genetic algorithms, ant colony optimisation, and simulated annealing. These methods are particularly useful when the objective is to obtain a good solution

quickly rather than to prove global optimality.

The choice between exact and heuristic approaches should therefore follow the structure of the problem. A complete Technician Routing and Scheduling Problem includes job sequencing, travel between visits, time windows, skills, and sometimes tools, spare parts, and crew synchronisation. Such a formulation can require a metaheuristic or matheuristic approach. The present thesis uses a narrower Field Technician Scheduling Problem. It focuses on daily technician assignment, skill compatibility, capacity restrictions, and base-to-job travel estimates. This more limited scope makes a CP-SAT formulation appropriate, while avoiding the complexity of fully optimised job-to-job routing.

A further way to manage computational complexity is horizon decomposition. Instead of solving the entire planning period as one large optimisation problem, the horizon can be divided into smaller sequential subproblems. Glomb et al. [14] study rolling-horizon optimisation and show how decisions made in one period affect the options available in later periods. This approach creates a trade-off: each daily problem can be solved more efficiently, but the resulting monthly plan is not guaranteed to be globally optimal. In the present prototype, the 30-day horizon is therefore solved day by day, while multi-day reservations transmit important commitments to subsequent days.

Recent work also shows that technician scheduling models can be used for more than generating a single schedule. Gamst and Pisinger [9] use the Technician Routing and Scheduling Problem as a decision-support tool for comparing scenarios involving workforce size, skills, working hours, and digitalisation. This perspective is particularly relevant to this thesis. The aim is not only to find a feasible assignment plan, but also to use the model to identify capacity bottlenecks and test the effect of alternative operational assumptions.

2.3 Machine Learning for Service-Duration Prediction

The quality of a technician schedule depends on the quality of the duration estimates used to construct it. If a job is expected to take less time than it actually requires, the scheduler may assign more work than a technician can realistically complete. This can create capacity problems, missed SLA windows, and overtime that was not anticipated during planning. For this reason, estimating service duration is treated in this thesis as a supervised regression problem: historical job and technician information is used to estimate the time required for a future intervention.

Machine Learning is widely used in industrial maintenance, although many studies focus on related but different prediction targets. Carvalho et al. [15] review Machine Learning applications in predictive maintenance, including fault detection, remaining useful life, and maintenance decision support. These applications help organisations understand the condition of an asset or the likelihood of failure. Duration prediction addresses a more

specific operational question: once a maintenance intervention has been identified, how long is it likely to require? This distinction is important because the predicted duration is used directly as an input to technician capacity planning.

Research in service management also shows that contextual information can improve operational predictions. Zuev et al. [16] study Machine Learning in IT service management, using ticket characteristics and historical resolution information to identify the risk of SLA violations. Although their application concerns IT incidents rather than physical field service, the study shows that job categories, past performance, and service-process information can support operational decision-making. In a field service setting, similar information may include equipment characteristics, job complexity, technician experience, travel requirements, and service priority.

The study by Poutiainen [17] is particularly relevant to this thesis because it focuses directly on field-service duration estimation and workforce qualification. The work combines structured information with textual task descriptions and shows that a data-driven approach can improve duration estimation compared with a historical baseline. However, its reported prediction results concern daily bundles of tasks rather than individual interventions. This difference matters because a model may perform differently when predicting the duration of a single job compared with the total workload of a technician during a day. Nevertheless, the study supports the central idea that fixed duration estimates can be improved by considering the characteristics of both the task and the workforce.

Random Forest is one possible method for this type of prediction. Introduced by Breiman [18], Random Forest combines the predictions of many decision trees and can represent nonlinear relationships and interactions between variables. These properties are useful when duration may depend on several connected factors, such as equipment age, job complexity, travel time, and technician experience. In the present thesis, Random Forest is used as a flexible regression model whose predictions and feature-importance summaries can be inspected rather than as an assumption that a more complex model will always perform better. Its performance is compared with the original static estimate and simpler alternative models.

The purpose of the predictive layer is therefore not to replace the dispatcher or to make a complete operational decision on its own. Its role is to provide a more informed estimate of service effort for the scheduling model. The following section explains how these estimates are connected to the CP-SAT optimiser and why predictive accuracy must be considered separately from the quality of the resulting schedule.

2.4 The Predict-Then-Optimise Paradigm

Predict-then-optimise is an approach that connects a predictive model with an optimisation model. In the first phase, a Machine Learning model estimates an uncertain quantity,

such as cost, demand, travel time, or service duration. In the second phase, an optimisation model uses that estimate to select a feasible decision. In this thesis, the predicted quantity is job duration and the downstream decision is the assignment of technicians to service jobs.

This approach is useful because operational decisions often depend on information that is not known with certainty at the time of planning. However, a prediction model with a low average error does not automatically produce the best operational decision. A small duration error may have little effect if a technician still has enough available capacity, while a similar error may make another job infeasible or cause an SLA-related delay. The quality of the prediction and the quality of the scheduling decision are therefore related, but they are not the same thing.

Elmachtoub and Grigas [10] formalise this distinction through the Smart Predict-then-Optimise (SPO) framework. Their work compares the decision made using predicted values with the decision that would have been made if the true values were known. This shifts attention from prediction error alone to the operational consequences of prediction error. The framework is important because it explains why a model should not be evaluated only through measures such as Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE) when its predictions are later used in an optimisation problem.

Applying decision-aware learning can be difficult when the downstream optimisation problem is complex. Mandi et al. [11] examine this challenge for hard combinatorial optimisation problems and discuss approaches such as relaxations and warm starts to reduce the computational burden of solving an optimiser repeatedly during model training. Their work shows that integrating prediction and optimisation more closely is possible, but it can require substantially more computation and a more specialised training process.

The present thesis adopts a sequential predict-then-optimise design. A Random Forest model is trained on historical service records and produces duration estimates for the simulation jobs. These estimates are then passed to the CP-SAT scheduling model, where they affect daily capacity consumption, labour cost, assignment feasibility, and multi-day reservations. The approach does not train the predictor directly on the final scheduling objective. Instead, it evaluates the predictive model and the scheduling model separately, making it possible to identify whether a change in planning results comes from duration estimation or from the optimisation constraints.

This separation is also important when interpreting the results. The predictive model is evaluated using historical duration outcomes, while the scheduling experiments compare plans generated under different duration assumptions. Because the simulation jobs do not contain realised future execution outcomes, the reported planning indicators describe the behaviour of the model under each duration regime rather than proven operational savings. This provides a clear basis for the experiments while identifying decision-aware learning and execution-based validation as directions for future work.

2.5 Decision Support and Digital-Twin Concepts in Field Service Management

Optimisation models are more useful in practice when planners can understand and compare their results. A Decision Support System (DSS) does not simply produce one recommended solution. It helps users inspect alternatives, identify constraints, and understand the trade-offs behind a decision. In field service operations, this may involve comparing technician assignments, travel requirements, workload distribution, service priorities, and unassigned jobs. Gupta et al. [19] discuss the role of artificial intelligence in operations-research decision-support systems, emphasising the use of analytical models to support, rather than replace, managerial judgement.

The user interface is therefore part of the decision-support process. A dispatcher may need to inspect the assignment of a particular technician, while a manager may be more interested in overall capacity shortages, service coverage, or the effect of a workforce reduction. Arin et al. [20] propose a Field Service Management information-system prototype with separate views for managers and field workers. Their work is relevant because it shows that different users require different types of information, even when they interact with the same operational data.

Digital Twin concepts extend this idea by linking a digital representation of an operation with information about its physical counterpart. In industrial maintenance, this can involve equipment data, workforce information, predictive models, and planning decisions. Gómez Fernández and Crespo Márquez [21] discuss how artificial intelligence and Machine Learning can support maintenance planning, while also highlighting practical issues such as model deployment, explainability, and changes in operational data over time. These issues are important for a future industrial implementation, where a duration model would need to be monitored and updated as equipment, technicians, and working practices change.

Some field-service and routing studies also consider dynamic decisions. For example, Park et al. [22] examine a waiting strategy in a vehicle-routing setting, where delaying a decision can create flexibility for future requests. Such approaches are relevant when live job arrivals, traffic conditions, or unexpected events require continuous rescheduling. However, they are outside the scope of the present prototype, which uses a synthetic planning scenario rather than real-time operational data.

The present system is therefore positioned as a digital-twin-inspired decision-support prototype. It provides a virtual representation of a synthetic field-service operation in which users can generate assignments, inspect a map and schedule board, compare duration assumptions, and test operational scenarios. It does not maintain a continuous connection with a physical service operation, and it is not intended to replace a production Field Service Management platform. Its purpose is to support scenario analysis and capac-

ity diagnosis before a decision is implemented in practice.

2.6 Research Positioning and Scope

The literature reviewed in this chapter shows that field service operations bring together several connected challenges: technician skills, service duration, travel, capacity restrictions, and changing operational conditions. Previous studies provide detailed routing formulations, heuristic and exact solution methods, predictive models, and decision-support applications. The present thesis does not claim to solve all of these challenges within one system. Instead, it combines selected elements that are directly relevant to daily technician assignment and capacity planning.

The research position of this thesis is a Field Technician Scheduling Problem in which Machine Learning is used to estimate service duration and Constraint Programming is used to allocate technicians to jobs. The model considers skill compatibility, daily capacity, base-to-job travel time, multi-day reservations, and crew requirements. This creates a more realistic planning problem than one based only on fixed job durations, while remaining smaller and more transparent than a complete Technician Routing and Scheduling Problem.

The study deliberately focuses on assignment rather than detailed route construction. Inter-job route sequences, exact start times, breaks, and live traffic updates are not optimisation decisions in the current prototype. This allows the model to examine how duration estimates and capacity restrictions affect assignment coverage without claiming to optimise every aspect of field service execution.

The contribution is therefore not a new optimisation algorithm or a deployed Digital Twin. It is the implementation and evaluation of a predict-then-optimise decision-support prototype for a defined synthetic field-service scenario. The evaluation compares alternative duration estimates, identifies capacity bottlenecks, and tests how operational disruptions affect the generated plans. Predictive accuracy is assessed using historical outcomes, while the scheduling experiments describe model-conditional planning results rather than realised operational savings.

This scope provides a controlled basis for studying the interaction between duration prediction, constrained technician assignment, and capacity diagnosis. It also identifies clear directions for future work, including detailed routing, live data integration, execution-based evaluation, and decision-aware learning.

Chapter 3

Methodology and System Architecture

This chapter describes the methodology and system architecture of the proposed field-service scheduling prototype. The approach follows a sequential predict-then-optimise process. First, a Random Forest model is trained on historical technician assignments to estimate the duration of future service jobs. These estimates are then passed to a CP-SAT scheduling model, which assigns technicians while considering skills, daily capacity, travel requirements, multi-day reservations, and crew requirements. Separating prediction from optimisation makes it possible to examine both the accuracy of the duration estimates and their effect on the schedules generated by the system.

The following sections describe the synthetic scenario and input data, the architecture of the prototype, the predictive and prescriptive layers, and the experimental design used for evaluation. Historical and simulation records are kept separate throughout the process so that future planning data are not used during model training.

3.1 Synthetic Data and Scenario Design

The prototype is evaluated using a synthetic but operationally plausible field-service scenario. It represents a mid-sized European industrial service provider operating across Italy, Germany, France, Spain, and the United Kingdom. The scenario includes technicians with different skills and experience levels, service jobs with different priorities and duration requirements, and travel information between technician bases and customer locations. It provides separate historical records for training the predictive model and future simulation jobs for the scheduling experiments.

The purpose of the synthetic scenario is to create a controlled setting in which duration estimates, capacity restrictions, and operational disruptions can be compared consistently. It is designed to support the evaluation of the proposed methodology rather than to reproduce the data of a specific company. The following subsections explain the rationale for this choice, the structure of the datasets, and the geographic and temporal characteristics of the scenario.

3.1.1 Rationale for Synthetic Data

Operational Field Service Management data often contain commercially sensitive information and personal data. No proprietary dataset was available for this study. Synthetic data were therefore created to build a reproducible research scenario while preserving the main characteristics needed for technician scheduling. The operational interpretation of the scenario was informed by the author's experience at Körber Tissue S.p.A. and OVS, but the data were constructed specifically for this thesis.

The synthetic tables encode controlled assumptions about job complexity, expected and actual duration, technician skills, priorities, geographic locations, and travel. This makes it possible to compare alternative duration regimes and planning scenarios under the same conditions. In particular, the separation between historical and simulation records allows the prediction model to be trained on past assignments and then applied to future jobs without using future information during training.

The use of synthetic data also creates limitations. The patterns learned by the prediction model depend on the rules used to generate the historical records, and the reported planning results describe the behaviour of the prototype within this scenario. Before the approach could be used for operational decisions in a real company, the assumptions, data quality, and model performance would need to be reassessed using real field-service data.

3.1.2 Input Datasets and Their Relationships

Five input datasets support the prediction and scheduling layers: technicians, historical jobs, historical assignments, simulation jobs, and a simulation travel matrix. The historical datasets provide the information used to train and validate the duration model, while the simulation jobs and travel matrix are used to generate future schedules. Their keys and join relationships are shown in Figure 3.1.

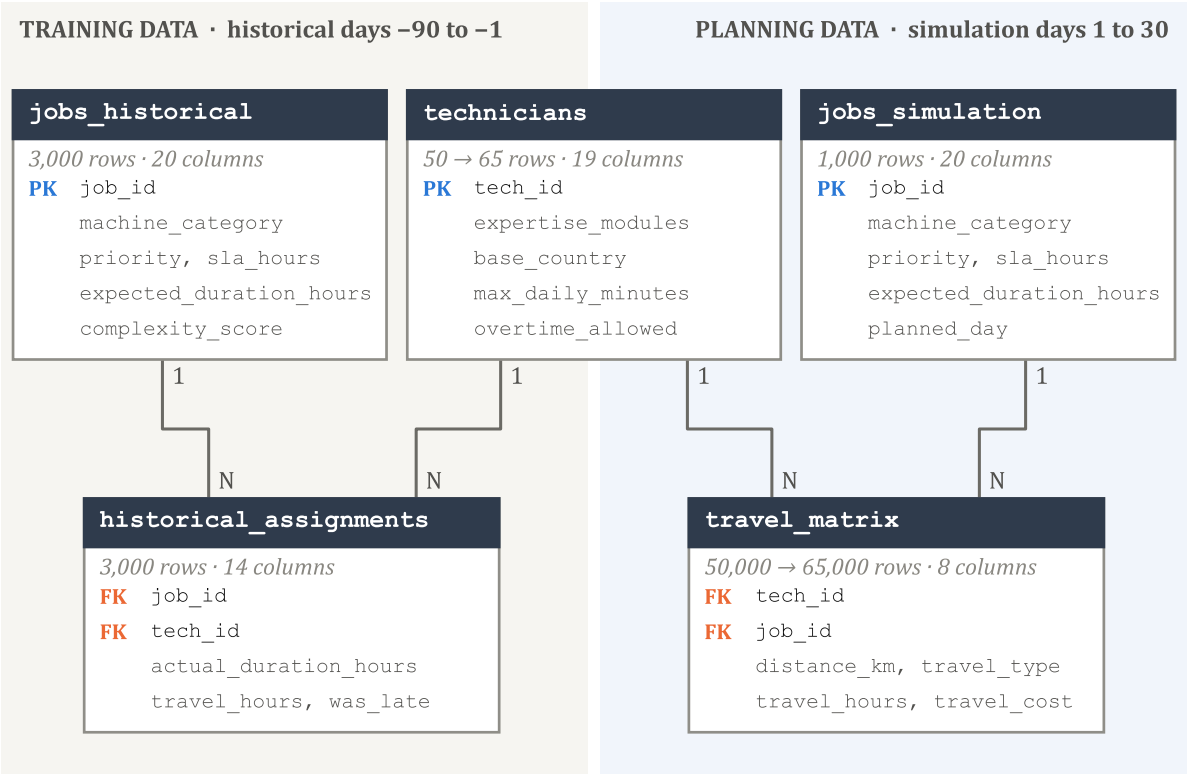


Figure 3.1: Five input datasets and their join relationships. Historical assignments link training jobs to technician attributes and provide observed travel hours. The separate travel matrix supplies simulation job–technician pairs. PK denotes a primary key and FK a foreign key.

Source: own elaboration from the five input datasets and prototype schema.

The **technicians dataset** contains 50 technicians distributed across five countries: Spain (14), Germany (13), Italy (9), France (9), and the United Kingdom (5). Each record includes capability information, such as job role, technical level, and eight expertise modules. It also includes performance information, with a mean experience level of 8.6 years, a mean quality score of 4.02/5, and a mean first-time-fix rate of 81.9%. Cost and availability information include a mean hourly cost of EUR 47, a daily capacity of 480 minutes, and overtime eligibility for 33 of the 50 technicians.

The job data is divided into separate **historical** and **simulation** tables to preserve temporal separation. The historical jobs table contains 3,000 records from Days -90 to -1 and includes the planned job attributes and expected duration. The historical assignments table records the observed execution outcomes, including actual duration, travel hours, SLA outcome, delay reason, and customer feedback. The simulation jobs table contains 1,000 records from Days 1 to 30. It includes expected durations but no actual execution outcomes, since these jobs are used for future planning. The simulation jobs cover six machine categories and three priority tiers, and 22.8% require two technicians. The corresponding proportion in the historical data is 25.0%.

The **travel matrix** pre-computes all technician–job pairings used by the scheduling model: 50,000 pairings for the initial workforce and 65,000 after workforce augmentation. For the original 50-technician workforce, the average distance is 1,140 km, the average travel time is 3.2 hours, and the mean direct travel cost is EUR 281 per trip. Transport mode is derived from distance: flights are used for routes above 800 km (85.0%), rail for distances between 300 and 800 km (12.2%), and road below 300 km (2.8%). Each pairing also includes a stochastic arrival-delay risk, with a mean of 39.6% and a range from 0.6% to 60.3%. This risk contributes to the composite risk-scoring framework. A binary `visa_required` variable represents cross-border administrative requirements.

The historical assignments table also provides an initial descriptive view of the duration bias motivating the predictive layer. Mean actual duration is 14.1 hours, compared with a mean planned duration of 12.3 hours. This corresponds to an average overrun of 1.8 hours, or approximately 15% of the planned duration. The overall late rate is 21.1%, although it differs substantially by priority: 85.1% for critical jobs, 34.3% for high-priority jobs, and 0.2% for normal-priority jobs. Among the 632 late assignments, missing parts account for 15.3%, travel delays for 13.1%, unexpected complexity for 7.6%, and customer unavailability for 5.1%.

3.1.3 Geographic Representation and Dataset Scale

Technician bases and job sites are distributed around European manufacturing and logistics hubs, including Milan, Frankfurt, Madrid, and Barcelona. Their coordinates provide a geographic representation of the cross-border operating scenario and are used for map visualisation and descriptive geographic analysis. Great-circle distances calculated from these coordinates are used in the geographic diagnostic reported in Table 4.2.

The scheduling model uses a separate stored travel matrix. This matrix contains synthetic distances, travel times, and travel costs for each technician–job pairing, and its distance values do not reproduce the great-circle distances calculated from the coordinates. Therefore, the coordinate-based analysis describes the spatial layout of the scenario, while all scheduling runs use the travel times and costs supplied by the travel matrix. This distinction limits how precisely the geographic results can be interpreted and should be considered when comparing map-based diagnostics with optimisation outputs.

The dataset scale consists of 50 technicians in the initial workforce, 3,000 historical records, and 1,000 simulation jobs. The workforce is expanded to 65 technicians in the augmentation scenario. This scale provides a manageable prototype instance while retaining heterogeneous skills, geographically dispersed demand, and two-technician crew requirements. The daily CP-SAT solver is run with a five-second time limit in the reported configuration.

3.2 System Architecture and Prototype Design

The prototype brings together data storage, duration prediction, technician scheduling, and visual inspection within a web-based decision-support system. Its architecture separates these functions so that the predictive model, scheduling model, and interface can be examined independently. This separation also supports the experiments: alternative duration inputs can be passed to the same scheduling model, and the resulting plans can be compared through the same interface.

3.2.1 Overall System Architecture

The system is implemented as a three-tier application, consisting of a presentation layer, an application layer, and a data layer.

The **presentation layer** is built with React 19 and TypeScript. It provides the scheduling controls, maps, assignment views, and scenario summaries used by the dispatcher. Leaflet supports geographic visualisation, while the schedule board includes drag-and-drop controls for manual reassignment. Socket.IO allows interface updates without a manual page refresh. These updates concern changes within the application; they do not imply that the prototype receives live data from a physical service operation.

The **application layer** uses FastAPI to handle application programming interface (API) requests. The Random Forest duration model, CP-SAT scheduling engine, and composite risk scorer are implemented as separate Python modules. Each module has a distinct role: estimating service duration, generating technician assignments, or calculating diagnostic risk indicators. Keeping these functions separate makes their behaviour easier to inspect and avoids combining prediction accuracy with scheduling performance.

The **data layer** uses PostgreSQL with the PostGIS spatial extension. It stores technician bases, job locations, and displayed travel paths alongside the operational records. The runtime database contains six tables: `technicians`, `jobs`, `historical_assignments`, `travel_matrix`, `routes`, and `job_assignments`. Historical and simulation jobs share the `jobs` table but are distinguished by a historical-record flag. The `routes` and `job_assignments` tables are populated when schedules are generated. Following workforce augmentation, the stored travel matrix contains 65,000 technician–job pairings.

The distinction between the five input datasets and the six runtime tables is therefore important. The input datasets provide the records needed to initialise the prototype, while the database also stores the assignments and displayed routes produced during planning.

3.2.2 Technology Stack and Design Choices

The technology stack supports a modular implementation using open-source tools. React provides the interactive interface, FastAPI connects that interface to the Python analytical modules, and PostgreSQL with PostGIS stores both operational and geographic infor-

mation. This arrangement allows the planning workflow to be presented through a web interface without embedding the analytical logic directly in the frontend.

Google OR-Tools CP-SAT is used to express binary technician–job assignments, integer capacity restrictions, and penalties for unmet staffing demand. It also reports solver status and bounds, allowing a feasible solution to be distinguished from a proven optimum for the daily formulation [13]. Its use is consistent with the focus on constrained assignment rather than complete route optimisation.

Random Forest provides the duration predictions used by the scheduling engine. It can represent nonlinear relationships and interactions among the input variables, while impurity-based feature importance offers a summary of how the fitted model uses those variables [18]. These summaries support model inspection but are not interpreted as causal effects. The choice of Random Forest does not imply that it must outperform simpler predictors; that question is examined through the predictive benchmark.

Together, these components keep data management, prediction, optimisation, and presentation separate while allowing them to operate within one prototype.

3.2.3 Prototype Development Process

Development followed four activities: requirements definition, design, construction, and local demonstration. The starting point was the dispatcher workflow: select a planning horizon, generate assignments, inspect the resulting plan, and compare alternative duration or disruption scenarios. The research requirements also included maintaining the separation between historical and simulation records and making unassigned demand visible.

The design translated these requirements into separate analytical modules and interface views. The overview presents aggregate planning indicators, while the map, timeline, schedule board, and job explorer allow users to inspect individual assignments. The analysis view brings together duration-model diagnostics and scenario controls. The purpose was to make both the selected assignments and the remaining capacity problems accessible to the user.

Construction involved implementing these components, reviewing the code, and checking system behaviour through experiments. AI tools supported aspects of implementation, debugging, and document preparation, as described in Section 1.7. The author retained responsibility for the modelling decisions, experimental design, and interpretation of the results.

The final activity prepared the prototype for local demonstration of the planning workflow. This was not a production deployment or formal user-acceptance study. Authentication, operational monitoring, and acceptance testing with commercial dispatchers remain outside the present implementation, as discussed in Section 6.2.

3.2.4 User Interface and Decision-Support Functions

The interface contains six views, each supporting a different part of the planning workflow.

Overview brings together indicators for the saved plan, including job coverage, unassigned work, regional coverage, travel, cost, and route risk. It provides a starting point for identifying where the plan leaves demand uncovered.

Map displays technician bases, job sites, and route polylines on a Leaflet basemap. A day control allows users to inspect assignments across the planning horizon. The displayed lines connect stored locations; they are not optimised road itineraries.

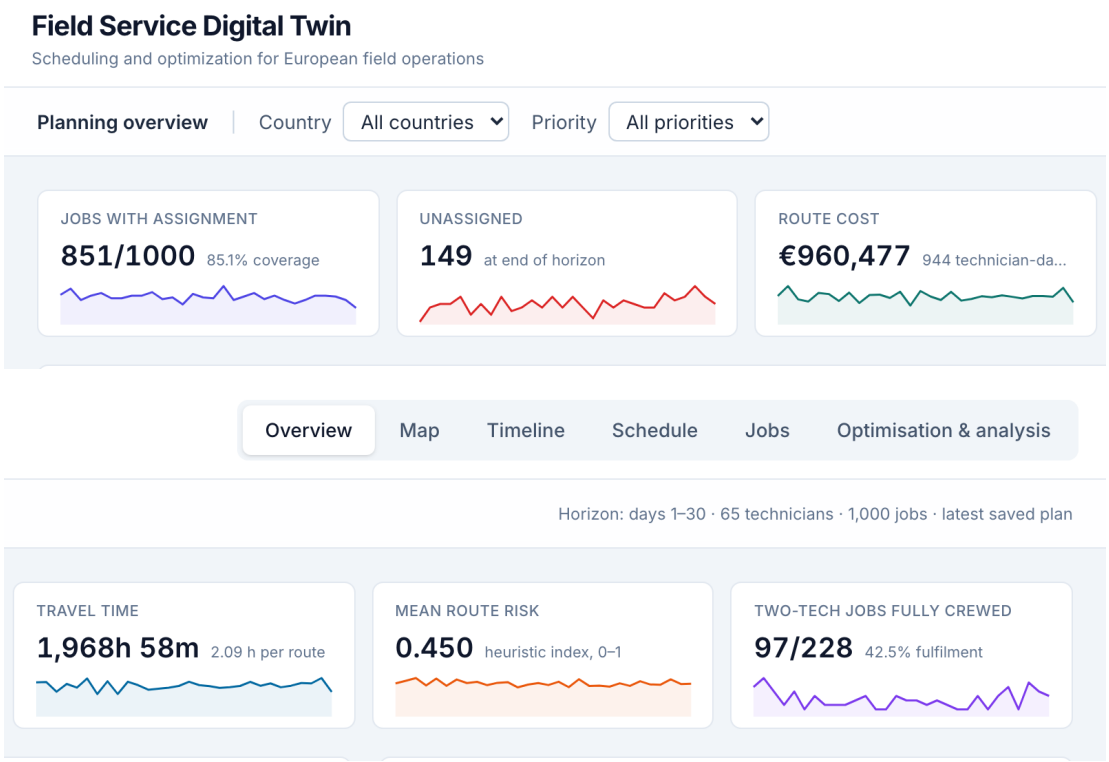
Timeline presents assignments by technician and day. Selecting an assignment opens a detail drawer containing the associated work, travel, cost, and jobs. The timeline is a visual reconstruction of saved planning information rather than a detailed execution schedule with optimised start times.

Schedule presents daily job cards and manual reassignment controls. These controls support interactive inspection and adjustment, subject to the validation boundary described in Section 3.6; their presence should not be interpreted as a guarantee that every manual change has the same validation as a solver-generated plan.

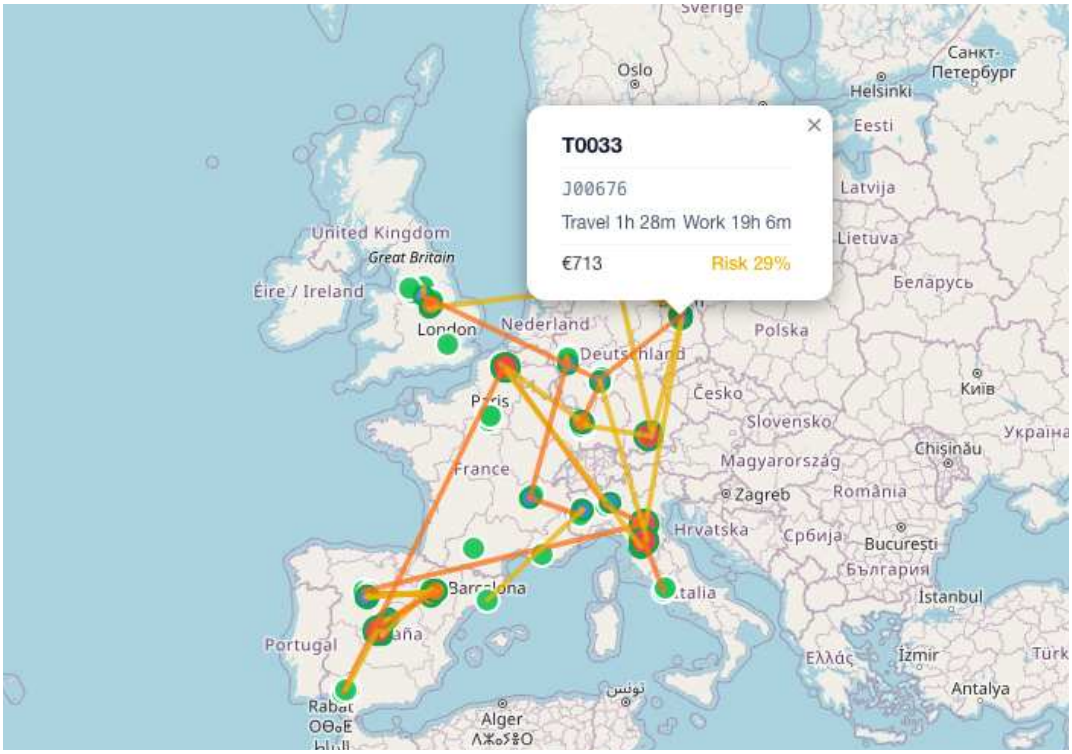
Jobs provides filters and job-level details, including assigned technicians, travel information, and risk indicators. It allows users to investigate individual jobs rather than relying only on aggregate results.

Optimisation & Analysis contains the duration-model panel, static and ML planning controls, and what-if scenario tools. This view supports comparisons between planning assumptions and helps users inspect how changes affect the generated assignments.

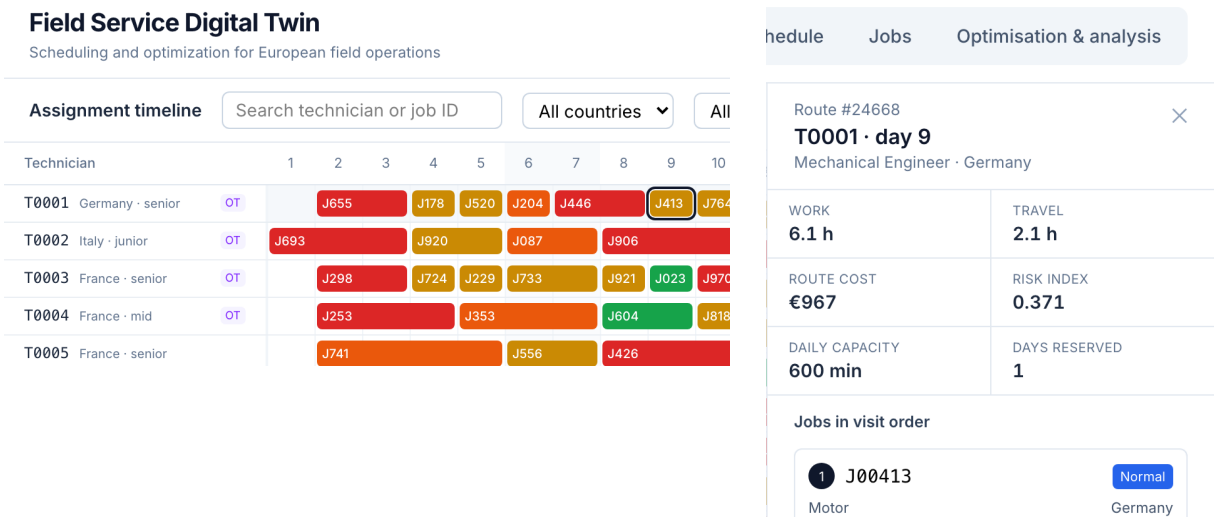
The selected interface states in screenshots 3.1 to 3.5 illustrate these functions. The numerical evaluation of the prototype is presented separately in Chapter 5.



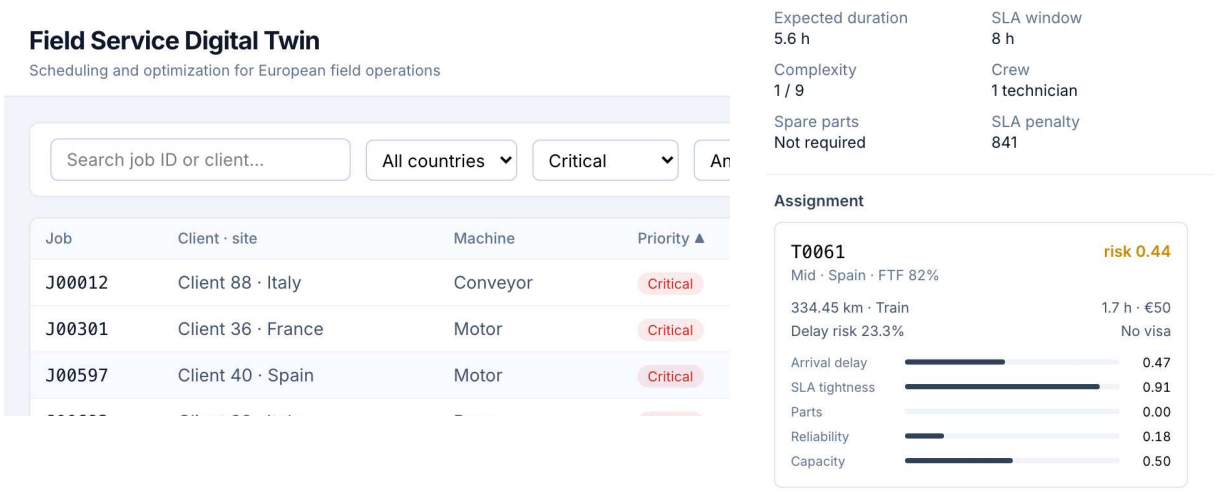
Screenshot 3.1: Overview, showing the six saved-plan indicators in two legible crops of the same screen: job coverage, unassigned work, route cost, travel time, route risk and two-technician crew fulfilment.
Source: author's screenshot of the FSM prototype.



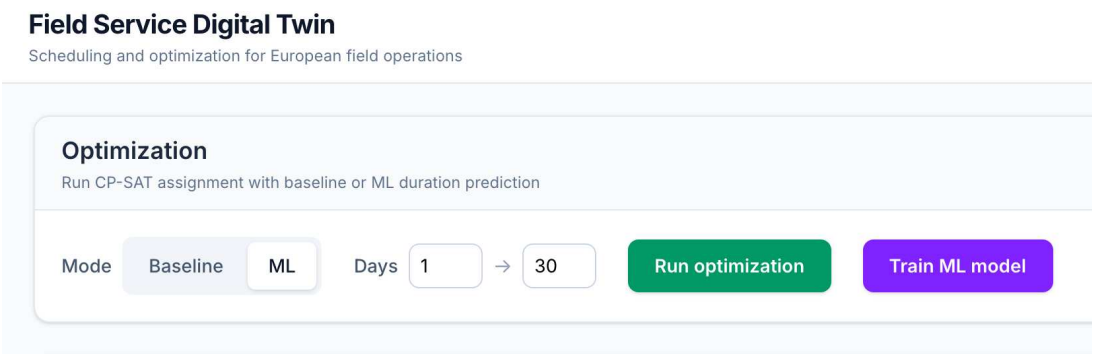
Screenshot 3.2: Map view with a selected route popup in static mode. The coloured lines connect stored locations rather than showing road itineraries.
Source: author's screenshot of the FSM prototype.



Screenshot 3.3: Timeline with a selected technician-day assignment (left) and its route detail drawer (right). Bar lengths are a visual reconstruction from saved route hours and daily capacity.
Source: author's screenshot of the FSM prototype.



Screenshot 3.4: Jobs view filtered to critical work (left), with the selected job's attributes, technician assignment and travel/risk detail (right). Expected duration is the static job input.
Source: author's screenshot of the FSM prototype.



Screenshot 3.5: Optimisation & Analysis with ML mode selected for the 30-day planning horizon. Model comparisons and saved planning indicators are reported in Tables 5.1 and 5.3.
Source: author's screenshot of the FSM prototype.

3.3 Predictive Layer: Service-Duration Estimation

The predictive layer estimates the service time required for a job when performed by a particular technician. It uses historical job attributes, technician profiles, and assignment records to learn the relationship between planned and actual duration. Its output is a duration estimate, not a technician assignment: the choice of technician is made subsequently by the scheduling model.

3.3.1 Prediction Problem and Target Variable

The prediction task is formulated as a supervised regression problem. Given the characteristics of a job and its assigned technician, the model estimates the `actual_duration_hours` recorded in the historical assignments dataset. These predictions are evaluated against the static `expected_duration_hours` field, which represents the original planning estimate.

The historical records show an average duration overrun of approximately 15% relative to the planned duration, as described in Section 3.1.2. However, this difference is not uniform across jobs. Interventions in the highest complexity tier, with a complexity score ≥ 6, exceed their planned duration by an average of 3.5 hours, compared with 0.6 hours in the lowest tier. This motivates examining whether contextual information can improve duration estimation beyond a fixed planning value.

A context-dependent predictor is therefore compared with both the raw static estimate and a calibrated multiplier. The multiplier provides a simple way to correct average bias, while the learned model can also respond to differences between jobs and technicians. This comparison helps distinguish the value of correcting systematic underestimation from the value of modelling individual variation.

Figure 3.2 shows how the predictive layer connects to technician scheduling.

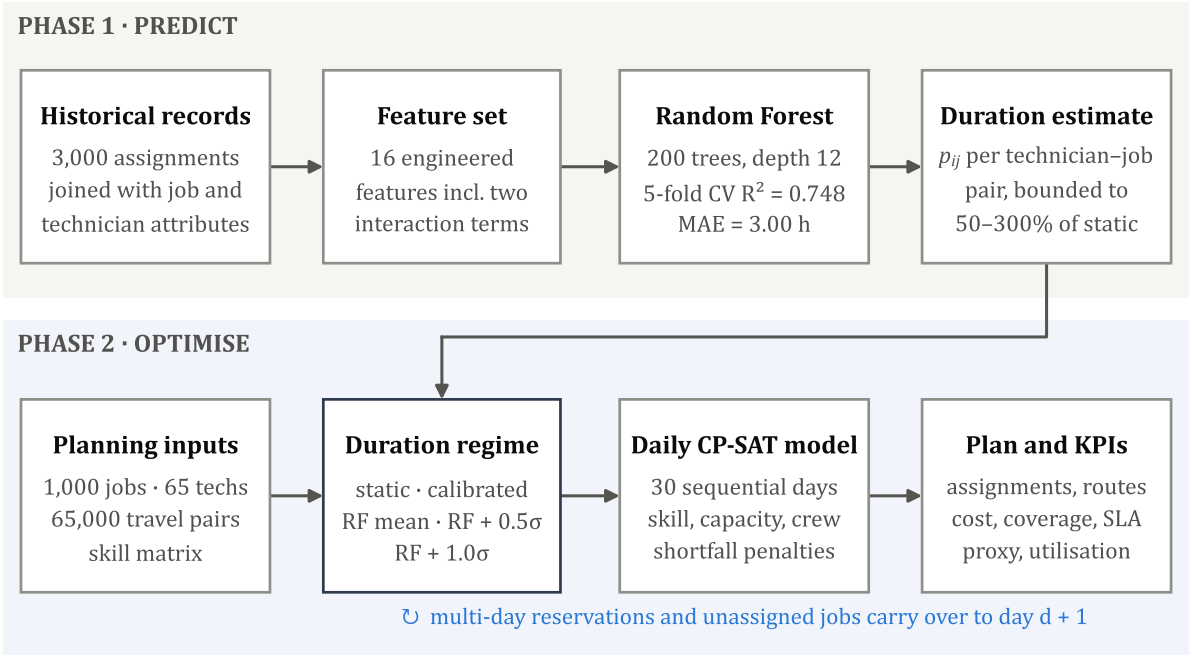


Figure 3.2: The two-phase pipeline. Phase 1 trains a Random Forest on 3,000 historical assignments joined with job and technician attributes; observed travel hours come from the assignments. Phase 2 uses predicted durations as input to a per-day CP-SAT solver. The ablation toggle selects static or ML-predicted durations for the optimiser.

Source: own elaboration from the prototype training and scheduling code.

3.3.2 Feature Engineering

The model uses sixteen features organised into four groups: job attributes, technician attributes, operational context, and interaction terms. Historical jobs and assignments are joined with technician records to construct these inputs. Historical travel hours come from the assignment records, while the simulation travel matrix supplies the corresponding planning information for future technician–job pairings.

Job attributes. The static estimate, `expected_duration_hours`, provides the main reference value. Additional inputs are `complexity_score`, measured on a 1–9 scale, `machine_age_years`, an ordinal priority encoding, and three binary intervention-type indicators: `is_corrective`, `is_installation`, and `is_upgrade`. These features allow the prediction to vary with the nature and difficulty of the intervention.

Technician attributes. The assigned technician is described through `experience_years`, `technical_level`, coded 0–2, `quality_score`, and `first_time_fix_rate`. Including technician information allows the same job to receive different duration estimates for different candidate technicians.

Operational context. The model also includes `travel_hours`, `spare_parts_required`, and `parts_in_stock`. These variables describe the logistical context associated with the intervention. Their inclusion allows the model to capture associations in the synthetic historical data; it does not establish a causal effect on service

duration.

Interaction terms. Two additional features combine job and technician information. The first, `complexity_x_age`, multiplies the complexity score by machine age. It represents the possibility that complexity and equipment age jointly affect service effort. The second, `experience_gap`, subtracts the task complexity score from the technician's years of experience. This is an engineered numerical proxy for the relationship between experience and task difficulty, rather than a direct measure of skill compatibility.

Skill compatibility is enforced separately by the scheduling model. The duration predictor estimates service effort for a pairing; it does not determine whether the technician holds the qualifications required to perform the job.

3.3.3 Model Selection

Random Forest is used as the implemented duration predictor because it can represent nonlinear relationships and interactions between the input features. It also provides feature-importance summaries that support inspection of the fitted model. The configuration uses 200 trees, a maximum depth of 12, a minimum leaf size of 5, and a random seed of 42.

The choice is evaluated against alternative models rather than treated as evidence that Random Forest is inherently superior. The benchmark in Table 5.1 compares Random Forest, linear regression, and gradient boosting using the same 5-fold cross-validation strategy. Linear regression achieves a slightly higher R^2 of 0.759, compared with 0.748 for Random Forest.

Random Forest is therefore retained as the prototype predictor, not as the most accurate model in the benchmark. It corrects the downward bias of the static estimate, while the simpler alternatives remain important comparators. The results are interpreted in terms of both aggregate prediction error and the extent to which improvement can be achieved through a simpler bias correction.

3.3.4 Validation Strategy

Predictive performance is evaluated using shuffled 5-fold cross-validation across the 3,000 historical assignments. Each fold is held out in turn, and the model is trained on the remaining records. RMSE, MAE, and R^2 are reported as mean scores across the five folds. Overall bias and residual spread are calculated from the pooled out-of-fold predictions.

Random Forest achieves an RMSE of 4.38 hours, an MAE of 3.00 hours, and a cross-validated R^2 of 0.748. The in-sample R^2 of 0.880 describes the fit to training records and is not used as evidence of performance on unseen assignments. The cross-validation scores provide the more relevant assessment of predictive fit within the synthetic historical population.

Historical and simulation records remain separate throughout this process. Simulation jobs are not used for training or validation. However, the shuffled cross-validation folds are not chronological or technician-grouped splits, so individual technicians may appear in both training and validation records. The evaluation therefore does not establish performance on entirely new technicians or under changing operational conditions.

After validation, the model is refitted on all 3,000 historical records for downstream inference. For simulation jobs, each predicted duration is bounded between 50% and 300% of the static expected duration. This limits extreme estimates before they enter the scheduling model; it is an implementation safeguard rather than a calibrated prediction interval.

3.3.5 Predictive Uncertainty and Safety Buffers

In the standard ML planning mode, the model supplies a single duration estimate for each job–technician pairing. The dashboard and CP-SAT solver use this point estimate without a job-specific uncertainty interval.

Two additional planning regimes introduce global safety buffers based on the cross-validation residuals. The pooled out-of-fold predictions produce a residual standard deviation of $\sigma = 4.38$ hours. Uniform offsets of 0.5σ and 1.0σ are added to predicted durations in the conservative regimes described in Section 3.5.2.

These offsets provide a controlled way to examine how greater planning conservatism changes the schedule. They do not represent calibrated confidence levels or guarantee that a particular proportion of jobs will finish within the buffered duration. The same offset is applied across jobs, although prediction uncertainty may differ by task or technician.

The spread of predictions across the Random Forest's 200 trees is also distinct from prediction error against held-out outcomes. Without calibration, tree-to-tree variation should not be interpreted as a reliable prediction interval. The present experiments therefore use the empirical residual spread as a global sensitivity parameter. Job-specific calibrated prediction intervals remain a possible extension, discussed in Section 6.3.

3.4 Prescriptive Layer: CP-SAT Technician Scheduling

The prescriptive layer uses the duration estimates to assign technicians to service jobs. Its purpose is to balance assignment cost against unmet staffing demand while respecting skill compatibility and daily capacity. The model considers travel from each technician's base to the job location, but does not optimise travel sequences between jobs. The 30-day planning horizon is solved through sequential daily subproblems, connected by job carryover and multi-day technician reservations.

3.4.1 Scheduling Problem and Mathematical Formulation

For each day d, the CP-SAT model determines which available technicians are assigned to the jobs released or carried over to that day. The duration input depends on the selected experimental regime: it may be the static estimate, a calibrated estimate, a Random Forest prediction, or a buffered prediction. The assignment formulation remains unchanged across these regimes.

Sets and parameters. For a given day d, let I_d denote the set of available technicians (indexed by i) and J_d denote the pool of newly released and carried-over jobs (indexed by j). Each job requires a crew of size $n_j \in \{1, 2\}$ and carries a priority weight $w_j \in \{3, 2, 1\}$ corresponding to critical, high, and normal urgency, alongside a base shortfall penalty of $P = 50{,}000$ EUR per unfilled crew slot. Skill compatibility between technician i and job j is given by the binary parameter $S_{ij} \in \{0, 1\}$. For each pair (i, j), let t_{ij} denote the base-to-job travel time in hours, c_{ij} the corresponding travel cost in euros, and p_{ij} the estimated repair duration in hours.

The notation p_{ij} accommodates technician–job-specific predictions. In ML mode, the estimate depends on both job attributes and technician characteristics. In static mode, the same job estimate is used across candidate technicians.

On the workforce side, each technician i has a standard or scenario-reduced shift duration b_i (480 minutes by default), a standard hourly wage h_i, an overtime hourly wage r_i, and an overtime eligibility indicator $o_i \in \{0, 1\}$. Let $e \in \{0, 1\}$ indicate whether overtime is enabled for the run. The effective daily capacity is $C_i = 600$ minutes when $eo_i = 1$, and $C_i = b_i$ otherwise.

Derived quantities. Total travel and service time is converted into integer minutes m_{ij}. The first-day capacity consumption a_{ij} is capped at the technician's effective daily capacity:

$$m_{ij} = \lfloor 60(t_{ij} + p_{ij}) \rfloor, \qquad a_{ij} = \min(m_{ij}, C_i). \tag{3.1}$$

The assignment cost q_{ij} combines travel and labour costs:

$$q_{ij} = c_{ij} + \begin{cases} h_i \dfrac{\min(m_{ij}, b_i)}{60} + r_i \dfrac{\max(0, m_{ij} - b_i)}{60}, & \text{if } eo_i = 1, \\ h_i \dfrac{m_{ij}}{60}, & \text{if } eo_i = 0. \end{cases} \tag{3.2}$$

This is the assignment-level cost used by the solver. It should be distinguished from the route-level accounting used to produce the dashboard's reported cost, as discussed in Section 6.2.

3.4.2 Constraints and Objective Function

The binary variable $x_{ij} \in \{0, 1\}$ indicates whether technician i is assigned to job j on day d. The integer variable $u_j \in \{0, \ldots, n_j\}$ records the number of unfilled crew slots. Assignment costs and penalties are expressed in integer cents for the CP-SAT objective:

$$\text{min} \quad \sum_{i \in I_d, j \in J_d} \lfloor 100 q_{ij} \rfloor x_{ij} + \sum_{j \in J_d} 100 P w_j u_j \tag{3.3}$$

$$\text{s.t.} \quad x_{ij} \leq S_{ij} \qquad \forall i \in I_d, j \in J_d \quad \text{(skill compatibility)} \tag{3.4}$$

$$\sum_{i \in I_d} x_{ij} + u_j = n_j \qquad \forall j \in J_d \quad \text{(crew demand)} \tag{3.5}$$

$$\sum_{j \in J_d} a_{ij} x_{ij} \leq C_i \qquad \forall i \in I_d \quad \text{(first-day capacity)} \tag{3.6}$$

$$x_{ij} \in \{0, 1\}, \quad u_j \in \{0, \ldots, n_j\} \qquad \forall i \in I_d, j \in J_d \quad \text{(variable domains)} \tag{3.7}$$

The objective combines assignment costs with priority-weighted penalties for unfilled crew slots. The large shortfall penalty encourages staffing coverage, with a greater penalty attached to critical and high-priority demand. It is a weighted objective, rather than a hard guarantee that every high-priority job will receive a complete crew.

Constraint (3.4) prevents assignments to technicians without the required expertise. Constraint (3.5) accounts for the required crew size through assigned technicians and unfilled slots. Because staffing gaps are penalised rather than prohibited, a two-technician job may receive only one technician when capacity is scarce. This distinction matters when interpreting assignment coverage: receiving at least one technician is not equivalent to receiving a complete crew. The formulation also does not enforce synchronised arrival times.

Constraint (3.6) limits each technician's first-day workload. For a job requiring more than one shift, the capped quantity a_{ij} consumes the full available shift rather than making the assignment infeasible. Subsequent occupied days are handled through reservations outside the daily solver, as explained below.

The objective does not contain a hard SLA-completion constraint or the composite risk score. SLA-related indicators and risk scores are calculated separately from the generated assignments.

3.4.3 Skill Compatibility and Capacity Feasibility

Skill eligibility is defined from the recorded expertise modules. Table 3.1 shows the mapping between machine categories and acceptable modules for the initial 50-technician workforce. Where alternatives are listed, matching at least one is sufficient; the OR operator is inclusive. Technicians may therefore belong to several overlapping skill pools. The literal module name used for robotic-arm work is `robots`.

An initial capacity diagnostic examines whether each simulation job could fit within a

Table 3.1: Skill compatibility matrix (initial 50-technician workforce).

Machine category	Required module(s)	Eligible technicians (n=50)
`pump`	pumps	28
`motor`	motors	15
`plc`	plc	19
`robotic_arm`	robots	15
`sensor_network`	sensors OR networks	35
`conveyor`	hydraulics OR drives OR motors	37

Source: own elaboration from the synthetic technician and job datasets.

single standard 480-minute shift. For this check, the entire service duration and the minimum stored travel time among eligible technicians must fit within the daily limit. This differs from the implemented multi-day formulation, which caps first-day consumption and reserves subsequent days.

The check identifies 762 of the 1,000 jobs as exceeding the single-shift limit, leaving 238 individually feasible jobs, as shown in Figure 3.3. Individual feasibility does not mean that all 238 jobs can be assigned together: they still compete for technicians and capacity. The diagnostic isolates the single-shift restriction before considering the combined scheduling problem.

The regional, daily, and skill-related implications of this bottleneck, together with the four-lever response, are examined in Section 4.6.

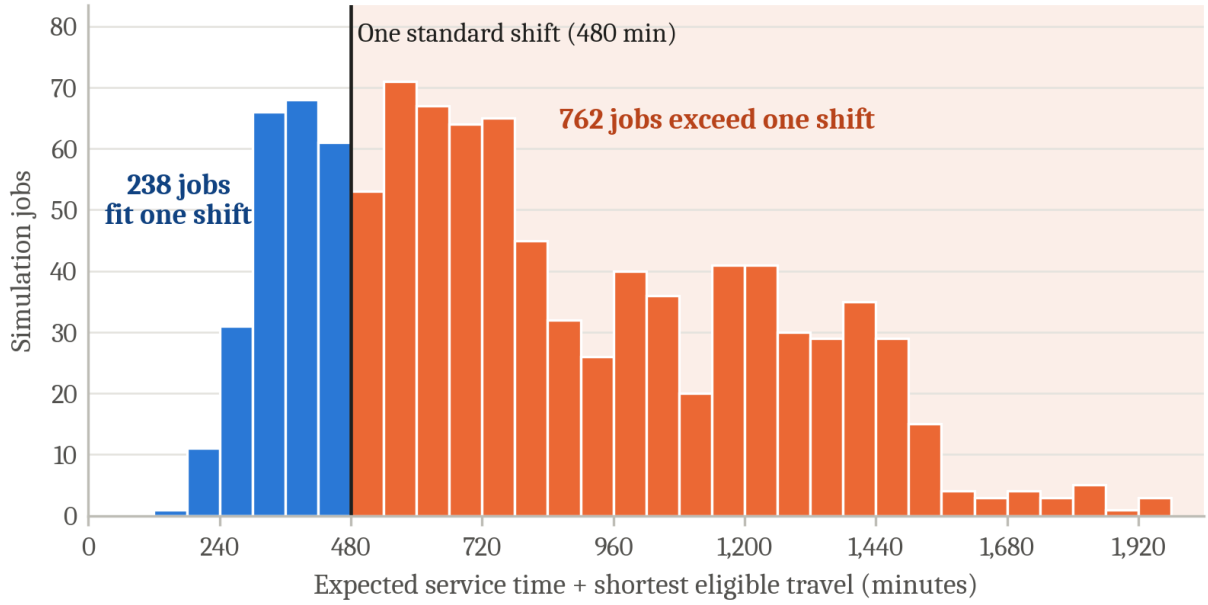


Figure 3.3: The capacity bottleneck. Of 1,000 simulation jobs, 762 (orange) exceed the 480-minute daily limit even when paired with the eligible technician with the shortest stored travel time in the original 50-person workforce. Only 238 jobs (blue) are feasible within a single shift under the original constraints.

Source: own elaboration from the synthetic simulation jobs and saved capacity diagnostic.

3.4.4 Daily Decomposition and Multi-Day Reservations

The 30-day horizon is divided into sequential daily subproblems to keep each solver invocation manageable. Daily decisions remain connected through technician reservations and the carryover of unassigned jobs.

After an assignment is selected, a technician whose travel and service requirement exceeds one shift is reserved for $\lceil (t_{ij} + p_{ij})/(C_i/60) \rceil$ working days, including the start day. The final blocked day is $d + \lceil (t_{ij} + p_{ij})/(C_i/60) \rceil - 1$. These are whole-day reservations: the prototype does not represent the precise start and completion times or release unused capacity on a partially occupied final day.

When daily capacity is insufficient, unassigned jobs are carried into the next day's candidate pool through the rolling carryover mechanism described as Lever 3 in Section 4.6. In the retained implementation, a two-technician job receiving one technician is counted as assigned and is not carried over to fill its missing second slot.

Daily decomposition reduces computational complexity but does not guarantee an optimal plan across the entire month. Each day's solution is conditional on earlier assignments and reservations. A full-horizon model could reconsider these commitments, for example by postponing non-urgent work to a less congested day. Carryover provides some flexibility, but does not remove this limitation.

Likewise, reservations do not enforce completion of every started job before the end of the 30-day horizon. The model represents assignment starts and coarse capacity commitments rather than a complete chronological execution schedule. Comparing this daily approach with a full-horizon formulation on smaller instances is identified as future work in Section 6.3.

3.4.5 Route-Display Post-Processing

After CP-SAT selects the daily assignments, a nearest-neighbour heuristic orders visits for technicians assigned to multiple jobs. The sequence is stored as a PostGIS `LINESTRING` geometry for display on the interactive map.

This post-processing does not change the assignments or replace the base-to-job travel inputs used by the optimiser. The displayed sequence should therefore not be interpreted as an optimised route or as evidence that job-to-job travel has been included in the scheduling constraints and objective.

3.5 Risk Scoring and Experimental Design

The experiments examine how the prototype responds to alternative duration estimates and changes in operational conditions. A composite risk index provides an additional diagnostic view of selected assignments. The duration ablation and what-if experiments then

compare plans generated under controlled changes to the inputs. These comparisons describe planning behaviour within the synthetic scenario, rather than realised execution performance.

3.5.1 Composite Risk Scoring

Each assignment is accompanied by a heuristic composite risk score intended to draw the dispatcher's attention to potentially vulnerable work. The score combines five normalised indicators: arrival-delay risk, SLA tightness, spare-parts availability, technician reliability, and capacity stress. The components and weights are shown in Table 3.2.

The index is a diagnostic ranking, not a calibrated probability of lateness or a repeat visit. Although it is displayed on a 0 to 100% scale, a score of 60%, for example, does not mean that the job has a 60% probability of failing. It indicates the combined level of pressure under the specified scoring rules.

Table 3.2: Composite risk score: components and weights.

Component	Weight	Formula	Interpretation
Arrival delay	0.30	$\min(r/50, 1)$; r in percentage points	Travel-matrix delay indicator, normalised to a score.
SLA tightness	0.25	$\min((t+p)/s, 1)$ for $s > 0$; otherwise 1	Travel t plus predicted service p relative to SLA hours s; excludes waiting.
Parts risk	0.20	1.0 / 0.2 / 0.0	Required but unavailable / required and stocked / not required.
Tech reliability	0.15	$1 - f/100$	f is first-time-fix rate in percentage points.
Capacity stress	0.10	$\min(u/100, 1)$	u is daily utilisation in percentage points.

Source: own elaboration from the prototype risk-scoring rules.

The weights define the heuristic scoring scheme used in the prototype; they are not fitted probabilities. The SLA-tightness component compares travel and service time with the SLA allowance but excludes waiting caused by carryover. The score therefore complements assignment and capacity indicators rather than measuring complete calendar-time SLA performance. It is not included in the CP-SAT objective.

3.5.2 Duration Regimes and Ablation Design

The primary duration comparison uses two operating modes. In static mode, the optimiser receives `expected_duration_hours`. In ML mode, it receives the Random Forest predictions. The workforce, candidate jobs, travel matrix, solver settings, constraint formulation,

and operational levers remain fixed within the comparison. This isolates the duration input as the intended experimental difference.

However, comparing the raw static estimate with Random Forest combines two effects: correcting average underestimation and capturing variation between technician–job pairings. A calibrated-static comparator is therefore included to examine how much can be achieved by a uniform correction alone.

The evaluation uses five duration regimes:

1. **Raw static estimates:** the original expected job durations.
2. **Calibrated static estimates:** the original durations multiplied by a single calibration factor to correct average bias.
3. **Random Forest point predictions:** duration estimates based on job, technician, and operational attributes.
4. **Random Forest with a 0.5σ buffer:** a uniform offset of half the out-of-fold residual standard deviation.
5. **Random Forest with a 1.0σ buffer:** a uniform offset of one out-of-fold residual standard deviation.

Comparing calibrated-static and Random Forest plans examines whether pair-specific predictions change scheduling beyond a general increase in duration. The buffered regimes examine the consequences of more conservative planning. They are sensitivity experiments, not calibrated guarantees about execution uncertainty.

Each regime is evaluated through the plans produced under its own duration inputs. Since the simulation jobs do not contain realised execution outcomes, differences in reported cost or SLA indicators cannot be interpreted directly as realised savings or improvements. The five-regime results are reported in Section 5.2.

3.5.3 What-If Scenario Simulation

The what-if module examines the response to changes in workforce availability, travel, spare parts, and daily capacity. Each experiment modifies selected inputs, reruns the daily scheduling process across the planning horizon, and compares the resulting planning indicators with the unmodified reference run.

Workforce reduction removes a configurable percentage of technicians from the active pool. This tests the sensitivity of assignment coverage to a smaller available workforce.

Travel disruption multiplies the stored travel times by a configurable factor. Longer travel requirements affect capacity consumption and may change which assignments remain attractive or feasible.

Parts supply disruption samples a specified share of jobs and sets `parts_in_stock` to false for sampled jobs that require parts. The percentage controls sampling, not the final stockout rate: some sampled jobs require no parts, while others are already out of stock. Parts availability is not a hard scheduling constraint. In static mode, this change affects the risk score without changing the duration input; in ML mode, it can also affect predictions through the parts feature.

Capacity reduction deducts a fixed number of minutes from the nominal daily shift. However, the effective capacity remains fixed at 600 minutes for overtime-eligible technicians when overtime is enabled. The experiment therefore does not reduce every technician's effective capacity equally.

These experiments test the implemented planning rules under specified disruptions. They are not an execution simulation of actual job completion. Single-factor and combined scenario results are reported in Section 5.3.

3.6 Verification and Evaluation Scope

Verification compares the implemented rules with the retained predictive and scheduling outputs. The predictive benchmark assesses error on held-out historical assignments. The scheduling review checks daily solver statuses, assignment counts and the aggregation of reported Key Performance Indicators (KPIs) against the saved records. It does not constitute an operational user trial.

Section 5.2 defines the planning indicators; Section 6.2 records the limitations of the execution model and accounting rules. The original scheduling experiments are retained throughout. A separately recorded predictive evaluation is identified in Section 5.1.2.

Chapter 4

Operational Diagnosis and Planning Implications

This chapter examines the operational problems revealed by the synthetic field-service scenario and explains the planning decisions they motivate. It brings together the distribution of jobs and technicians, historical duration and lateness patterns, prediction errors, and the initial scheduling results. The purpose is to understand where the main pressures arise before evaluating alternative planning configurations in Chapter 5.

Unless stated otherwise, the diagnostics refer to the initial workforce of 50 technicians. The workforce is later expanded to 65 technicians as part of the capacity intervention. The analysis moves from regional and daily workload imbalances to duration, travel, and skill constraints, before connecting these findings to the four-lever response. Throughout the chapter, descriptive indicators are used to identify questions for the scheduling model; they do not independently establish the operational or financial benefit of an intervention.

4.1 Geographic Distribution and Regional Capacity Imbalances

The initial workforce consists of 50 technicians distributed across Spain (14), Germany (13), Italy (9), France (9), and the United Kingdom (5). These technicians provide the capacity available for 1,000 simulation jobs over the 30-day planning horizon. Figure 4.1 shows the locations of technician bases and job sites. The main planning question is whether qualified capacity is available where the demand occurs, rather than whether the total workforce appears sufficient at an aggregate level.

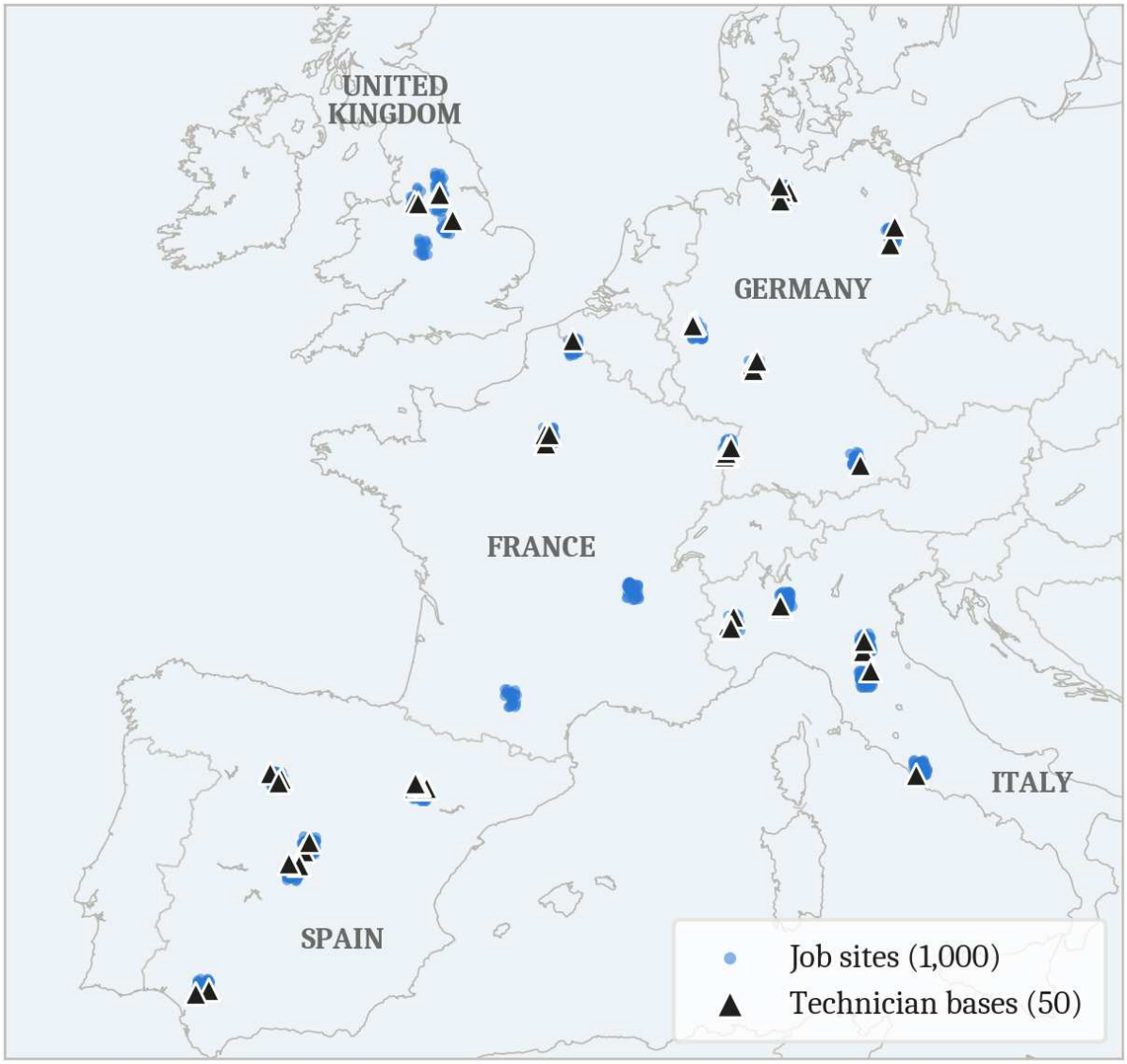


Figure 4.1: Geographic distribution of the dataset: technician bases (triangles) and simulation job sites (circles) across the five operating countries.

Source: own elaboration from the synthetic technician and simulation-job datasets.

A first assessment compares each country's expected service demand with the nominal capacity of its resident technicians. Demand is approximated by multiplying the number of jobs by the population mean expected repair duration, approximately 12.1 hours. Capacity is calculated as the number of resident technicians multiplied by 30 days and an eight-hour shift.

The resulting demand-to-capacity ratios are reported in Table 4.1. This calculation excludes travel, specific skill requirements, and additional crew members. It is therefore an initial screening of regional pressure, rather than a measure of feasible scheduling coverage.

Italy has the highest ratio, at 1.49, while Germany has the lowest, at 0.72. Across the entire scenario, expected service demand is approximately 12,090.1 hours, already exceed-

Table 4.1: Country-level coverage deficit.

Country	Technicians	Sim. jobs	Capacity (h)	Demand (h)	Demand / Cap
Italy	9	266	2,160	3,216	1.49
Spain	14	287	3,360	3,470	1.03
France	9	173	2,160	2,092	0.97
UK	5	89	1,200	1,076	0.90
Germany	13	185	3,120	2,237	0.72

Source: own elaboration from the synthetic technician and simulation-job datasets.

ing the 12,000 hours of nominal standard-shift capacity before travel and additional crew requirements are included.

These figures suggest that a uniform increase in workforce size would overlook important regional differences. Additional qualified capacity in Italy, or assignments drawing on technicians based elsewhere, are therefore relevant options to test. However, Germany's apparent surplus does not automatically translate into usable capacity for Italian jobs. The technicians must have the required skills, and the associated travel must be included in the assignment calculation.

Geographic proximity is examined more closely in Table 4.2, which reports the distance from each job to its nearest skill-compatible technician base. These are great-circle distances calculated from the coordinates. They are distinct from the stored travel-matrix distances used in the scheduling experiments.

Motor jobs have the largest mean distance to an eligible technician, at 143 km, followed by robotic-arm jobs at 87 km. In every category, the mean exceeds the median, indicating that more distant jobs increase the average. This makes the distribution of eligible technicians relevant alongside their total number.

Table 4.2: Great-circle distance to the nearest eligible technician, calculated from job and base coordinates for the original workforce.

Machine category	Mean nearest (km)	Median (km)	Sim. jobs
`motor`	143	43	128
`robotic_arm`	87	30	182
`plc`	78	27	187
`pump`	62	19	153
`sensor_network`	43	20	181
`conveyor`	42	17	169

Source: own elaboration from the synthetic job locations and technician bases.

A further illustration considers the effect of adding one standard-shift technician in each country separately. Each addition contributes 240 nominal hours over the 30-day horizon. Table 4.3 shows the resulting demand-to-capacity ratios, keeping the estimated service demand unchanged.

Table 4.3: Illustrative country demand-to-capacity ratios after adding one local technician.

Country	Techs (initial)	Ratio before +1	Ratio after +1
Italy	9	1.49	1.34
Spain	14	1.03	0.96
France	9	0.97	0.87
UK	5	0.90	0.75
Germany	13	0.72	0.67

Source: own elaboration from the synthetic country-level demand and workforce counts.

Italy remains above nominal capacity after one addition, with its ratio falling from 1.49 to 1.34. Spain's ratio falls from 1.03 to 0.96. These calculations help identify where local additions deserve further investigation, but they do not estimate jobs served, travel savings, or a financial return. Those outcomes depend on the skills and assignments of the added technicians.

Demand also varies over time. Expected service demand from newly released jobs exceeds the initial workforce's 400-hour standard daily capacity on 15 of the 30 days. This comparison again excludes travel and additional crew members. It motivates examining whether unassigned work can be carried into later days, while recognising that regional, skill, and multi-day commitments may restrict the capacity available then.

4.2 Duration Bias and Prediction Errors

The historical assignments reveal a systematic difference between planned and actual service duration. Across 3,000 records, actual duration averages 14.12 hours, compared with a static expected duration of 12.30 hours. The average overrun is therefore 1.82 hours, or 14.8% of the planned mean. Expressed as prediction error, the static estimate has a mean signed error of -1.82 hours and is approximately 12.9% below the actual mean.

The two percentages use different reference values: 14.8% expresses the overrun relative to planned duration, while 12.9% expresses the shortfall relative to actual duration. Figure 4.2 illustrates the difference and shows how it widens with job complexity.

This bias matters for capacity planning because a schedule constructed from underestimated durations can appear to accommodate more work than the duration evidence supports. Correcting the estimates may consequently reduce planned assignment coverage, even when the predictions become more accurate.

Random Forest brings the mean signed error of the held-out predictions to approximately zero, with an out-of-fold MAE of 3.00 hours. Removing average bias does not remove individual prediction errors. A model can be approximately correct on average while still underestimating some jobs and overestimating others. For this reason, the duration ablation includes a calibrated static multiplier as well as Random Forest, allowing the effect of a uniform bias correction to be distinguished from that of pair-specific predictions.

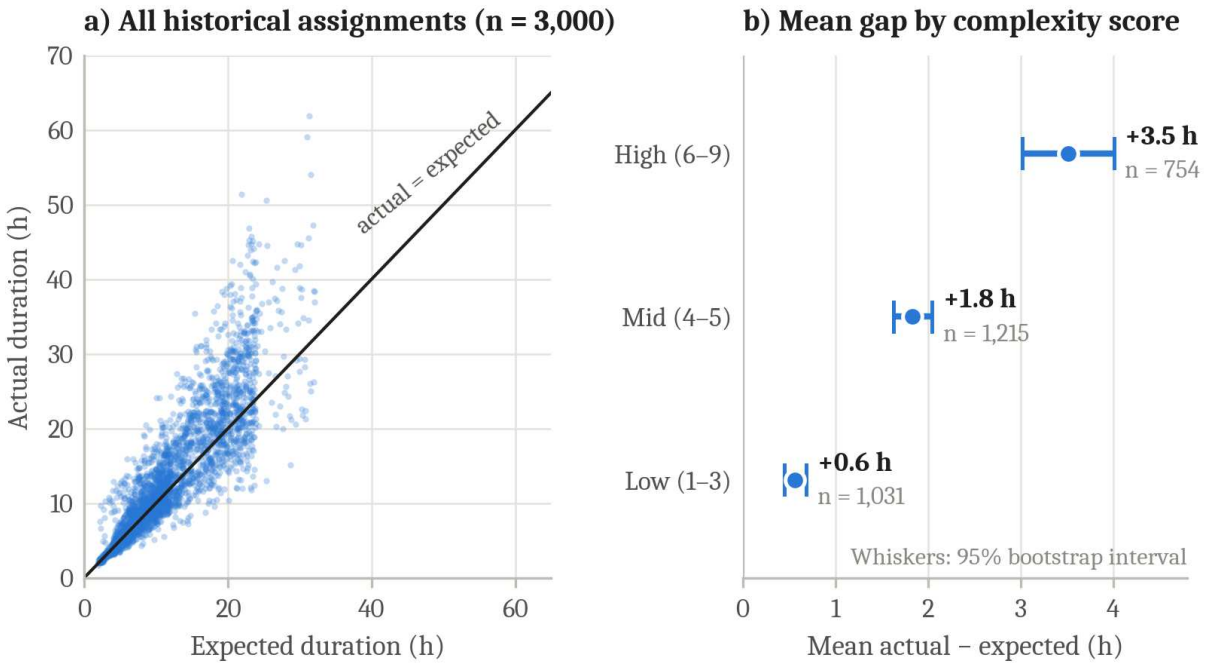


Figure 4.2: Planned and actual historical durations. Panel (b) reports paired mean gaps by complexity; whiskers are 95% bootstrap intervals from the assignments in each subgroup.

Source: own elaboration from the synthetic historical assignments.

Subgroup errors provide a more detailed view. Table 4.4 uses the separately saved, job-identified out-of-fold predictions described in Section 5.1.2. The pooled Random Forest MAE in this evaluation is 2.988 hours. Priority-level MAE ranges from 2.96 hours for critical jobs to 3.02 hours for high-priority jobs, a difference of about 2%. Larger contrasts occur between machine categories and countries: sensor networks have an MAE of 2.74 hours compared with 3.14 hours for robotic arms, while Spain records 2.79 hours compared with 3.15 hours for Italy.

Table 4.4: Random Forest out-of-fold MAE by subgroup in the separate job-identified predictive evaluation.

Subgroup dimension	Best subgroup	MAE (h)	Worst subgroup	MAE (h)
Priority	critical	2.96	high	3.02
Machine category	`sensor_network`	2.74	`robotic_arm`	3.14
Country	Spain	2.79	Italy	3.15

Source: own elaboration from the saved job-identified out-of-fold predictions.

In the same job-identified evaluation, assignments in the highest absolute-error decile have a historical lateness rate of 36.0%, compared with 12.7% in the lowest decile. The correlation between absolute residual error and the binary lateness indicator is 0.156. This is a descriptive association: it does not show that prediction error caused the recorded lateness.

The subgroup results help identify where further examination of features or uncertainty estimates may be useful. However, separate country and machine-category sum-

maries do not establish whether errors are concentrated in combinations such as critical robotic-arm jobs in Italy. Assessing those intersections would require enough observations in each joint category. The present evidence therefore motivates targeted investigation rather than an automatic rule for applying subgroup-specific buffers.

4.3 Historical SLA Lateness and Associated Factors

The historical dataset contains 632 late assignments out of 3,000. These are recorded historical labels, distinct from the planning-time SLA proxy used to evaluate generated schedules in Chapter 5. The late rate varies considerably by priority: 85.1% for critical jobs, 34.3% for high-priority jobs, and approximately 0.2% for normal jobs. These differences must be read in the context of the synthetic generator, where priority is closely linked to the contracted SLA window.

Figure 4.3 and Table 4.5 summarise the reasons recorded for late assignments. Missing parts account for 97 cases, travel delays for 83, unexpected complexity for 48, and customer unavailability for 32. The remaining 372 cases have no specific recorded reason. This unspecified category limits how much of the lateness pattern can be explained from the reason labels alone.

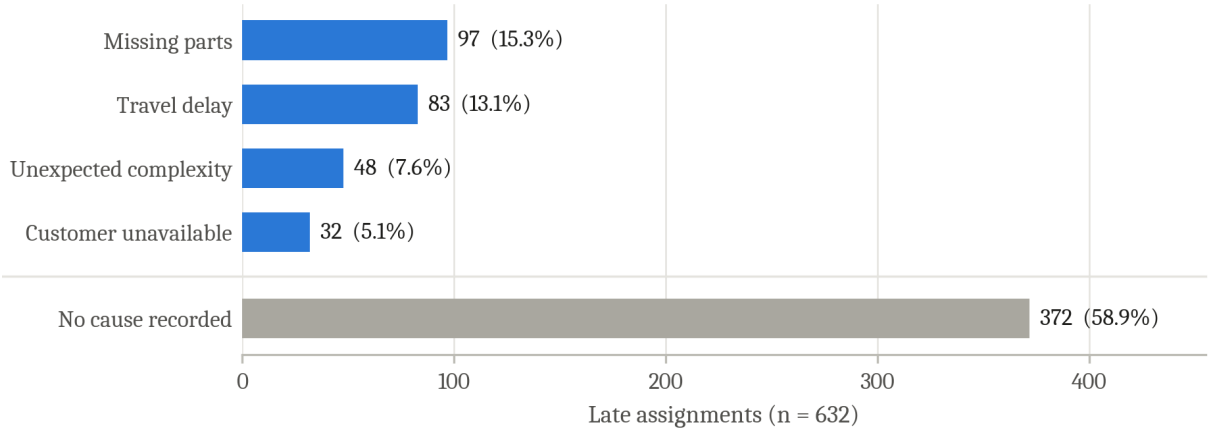


Figure 4.3: Recorded causes among the 632 late assignments.

Source: own elaboration from the synthetic historical assignments.

Table 4.5: Cause of lateness cross-tabulated by priority.

Delay reason	Critical	High	Normal	Total
Missing parts	43	52	2	97
Travel delay	44	39	0	83
Unexpected complexity	25	23	0	48
Customer unavailable	22	10	0	32
No specific cause	226	145	1	372
Total	360	269	3	632

Source: own elaboration from the synthetic historical assignments.

The country and machine-category breakdowns in Tables 4.6 and 4.7 show where the recorded cases are concentrated. Italy and Spain together account for 54 of the 97 missing-parts cases and 43 of the 83 travel-delay cases. Robotic-arm jobs have the largest number of missing-parts cases, at 21, followed by motor and pump jobs with 19 each.

These are counts among late assignments, not comparable rates of exposure to each problem. A country or machine category with more cases may also contain more jobs. Assessing relative incidence would require the relevant total number of jobs in each group.

Table 4.6: Cause of lateness cross-tabulated by country.

Delay reason	France	Germany	Italy	Spain	UK	Total
Missing parts	14	15	29	25	14	97
Travel delay	12	21	15	28	7	83
Unexpected complexity	7	8	9	15	9	48
Customer unavailable	9	5	2	14	2	32
No specific cause	58	65	92	115	42	372
Total	100	114	147	197	74	632

Source: own elaboration from the synthetic historical assignments.

Table 4.7: Cause of lateness cross-tabulated by machine category.

Delay reason	Conv.	Motor	PLC	Pump	Rob. arm	Sensor net.	Total
Missing parts	13	19	16	19	21	9	97
Travel delay	17	16	18	11	7	14	83
Unexpected complexity	7	11	6	13	6	5	48
Customer unavailable	6	7	8	3	5	3	32
No specific cause	69	56	49	59	75	64	372
Total	112	109	97	105	114	95	632

Source: own elaboration from the synthetic historical assignments.

To examine several factors jointly, a logistic regression relates the binary historical lateness label to six operational predictors. Contracted SLA duration is included instead of a priority dummy because priority and SLA duration are linked by construction, and very few normal-priority jobs are recorded as late. The stockout indicator equals one only when parts are both required and unavailable.

Table 4.8 reports the estimated odds ratios, confidence intervals, and p-values. Each odds ratio describes the association with lateness while holding the other included predictors constant.

Complexity and contracted SLA duration are the predictors associated with lateness at the reported significance threshold. Each additional complexity point corresponds to estimated odds multiplied by 2.03, while each additional SLA hour corresponds to a multiplier of 0.80. These are changes in odds, not direct changes in the probability of lateness.

Although missing parts and travel delays are the most frequent specified reasons

Table 4.8: Logistic regression of SLA lateness (`was_late`) on operational drivers. $N = 3{,}000$ assignments; McFadden pseudo-$R^2 = 0.66$; likelihood-ratio test $p < 0.001$.

Predictor	Odds ratio	95% CI	p-value
Complexity score (per point, 1–9)	2.03	1.82–2.26	< 0.001
Contracted SLA window (per hour)	0.80	0.78–0.82	< 0.001
Travel time (per hour)	1.06	0.99–1.14	0.121
Parts required but unavailable (0/1)	0.79	0.53–1.17	0.240
Cross-border assignment (0/1)	1.08	0.75–1.57	0.675
Machine age (per year)	0.99	0.96–1.01	0.215
Intercept	1.01	0.52–1.98	0.971

Source: own elaboration from the synthetic historical assignments.

among late assignments, the baseline stockout indicator and travel duration do not reach the 5% significance threshold in the fitted regression. The two analyses answer different questions: the reason counts describe recorded explanations among late jobs, while the regression examines conditional associations across all historical assignments. A non-significant coefficient does not establish that the corresponding operational problem is unimportant.

For a planner, these findings support closer inspection of complex jobs with tight SLA windows. They also motivate disruption scenarios involving travel and parts availability. However, the regression describes relationships within the synthetic historical data; it does not estimate how much lateness would fall following a particular intervention.

4.4 Travel Requirements and Their Planning Implications

The baseline travel matrix includes all possible technician–job pairings for the initial 50-technician workforce. Across this candidate pool, the mean distance is approximately 1,140 km, the mean one-way travel time is 3.2 hours, and the mean direct trip cost is EUR 281. These values describe the available pairings, including distant cross-border alternatives, rather than the journeys selected in the final schedule.

Transport mode is assigned through distance bands in the synthetic generator: flights are used above 800 km, rail for intermediate distances, and road transport for shorter trips. Figure 4.4 shows the resulting transport-mode and travel-duration distributions.

Travel influences the scheduling model through both time consumption and assignment cost. A qualified technician may therefore be an unattractive choice if reaching the job consumes too much capacity or adds substantial expense. This is why geographic coverage and skill availability need to be considered together.

The average trip cost of EUR 281 cannot be compared directly with the average hourly

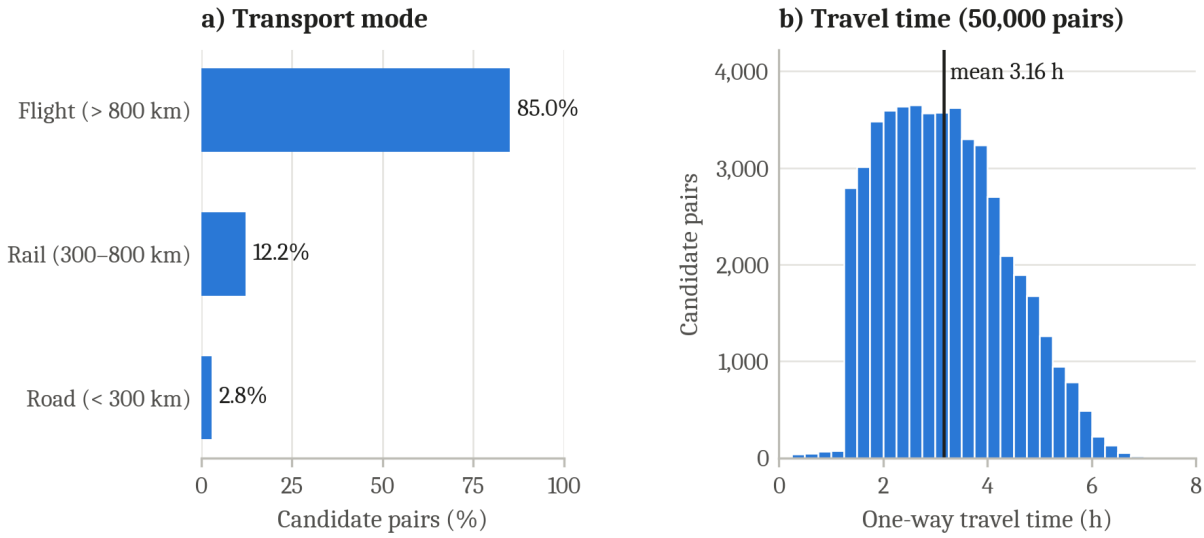


Figure 4.4: Travel mode distribution (left) and travel duration distribution (right).

Source: own elaboration from the synthetic travel matrix.

wage of EUR 47 to conclude which cost component dominates. One is a cost per journey and the other a cost per hour. A meaningful comparison requires applying wage rates to the relevant travel and service hours and examining the assignments actually selected by the solver.

The distance thresholds are also scenario assumptions. For example, lowering the flight threshold from 800 km to 600 km would reclassify some intermediate-distance pairings as flights. Evaluating a transport-policy change would require corresponding adjustments to the generated travel times and costs. A regional-base alternative would also require revised locations and travel inputs. These are possible extensions of the scenario, rather than policies whose benefits have already been established here.

The 15 technicians introduced during workforce expansion use slightly different travel-generation parameters from those used for the original workforce. Consequently, comparisons between the initial and expanded configurations reflect changes in travel inputs as well as headcount. This distinction is retained when interpreting the four-lever response in Section 4.6.

4.5 Job Demand and Skill Bottlenecks

The 1,000 simulation jobs are divided into normal (62.6%), high (22.3%), and critical (15.1%) priority tiers. Figure 4.5 shows these proportions alongside the distributions of complexity and machine category. Each category is linked to eligible technicians through the skill-compatibility rules in Table 3.1.

A simple indicator of skill pressure is the number of jobs in a machine category divided by the number of eligible technicians. For the initial workforce, this gives approximately 4.6 jobs per qualified technician for conveyors, 5.2 for sensor networks, and 5.5 for pumps.

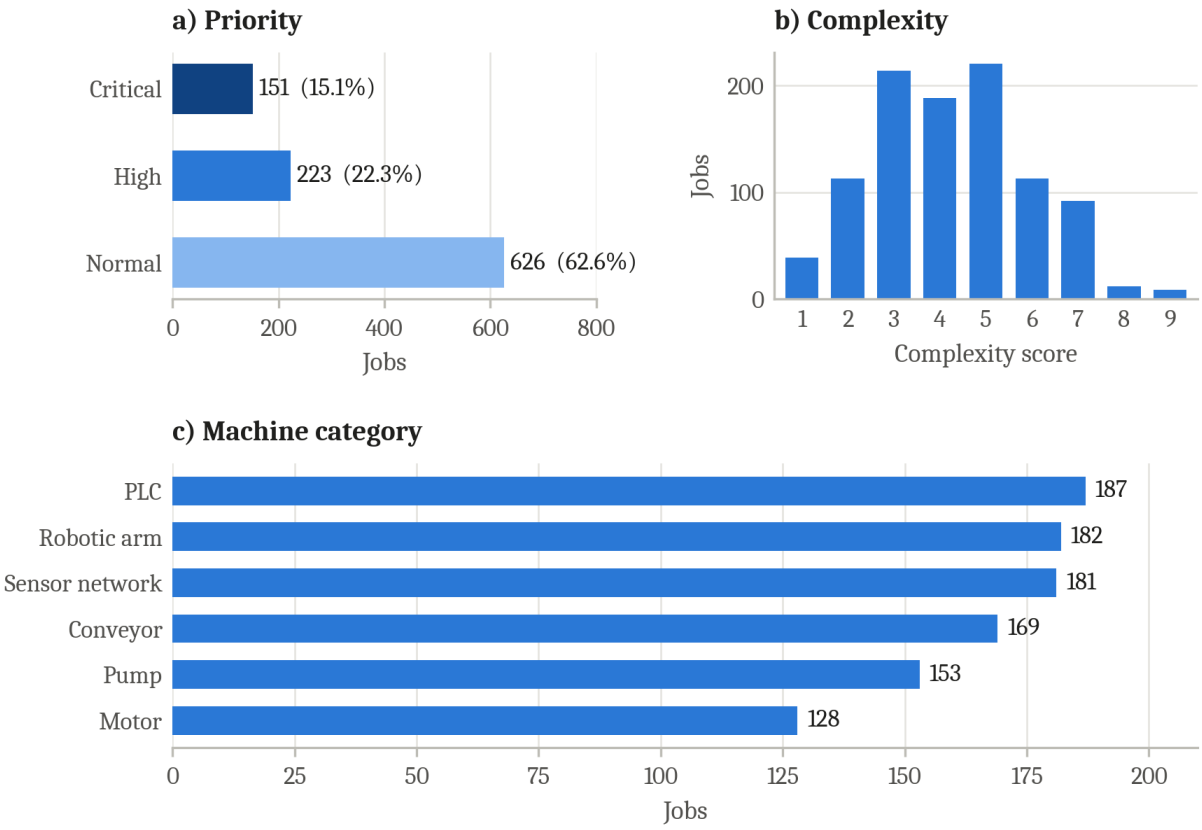


Figure 4.5: Simulation job counts by priority, complexity score and machine category.

Source: own elaboration from the synthetic simulation jobs.

The ratios are higher for motors (8.5), PLCs (9.8), and robotic arms (12.1).

These ratios identify robotic-arm, PLC, and motor qualifications as the most constrained pools by this measure. They do not account for service duration, additional crew members, or the fact that a technician may be eligible for several categories. They should therefore be used to select skill-expansion scenarios for testing, rather than as direct estimates of individual workload.

For illustration, adding four technicians certified for robotic-arm repairs reduces that category's ratio from 12.1 to 9.6 jobs per qualified technician. This improves the nominal relationship between demand and qualified headcount, but does not establish how many additional jobs can be assigned. Their location, availability, other qualifications, and the duration of the jobs also matter.

The planning implication is that workforce expansion needs to consider the composition of skills, not only the total number of technicians. Because qualifications overlap, adding capacity in one category may also change the assignments available elsewhere. The scheduling model is needed to evaluate those effects across the full job pool.

4.6 Capacity Bottlenecks and the Four-Lever Response

The initial scheduling run provides a direct indication of the capacity problem: under the single-shift configuration, only 198 of the 1,000 jobs receive an assignment. The diagnostic in Figure 3.3 helps explain this result. For 762 jobs, expected service duration plus the minimum eligible stored travel time exceeds the 480-minute limit.

This restriction excludes most of the long repairs before competition for shared technician capacity is fully considered. It also explains why simply searching harder within the same daily formulation would not resolve the problem. Jobs that cannot fit within the permitted shift need a different representation of their capacity requirements or a change in available capacity.

The response combines four changes: adding 15 technicians, enabling a 600-minute daily cap for overtime-eligible staff, reserving technicians across multiple days with rolling carryover of unassigned jobs, and applying priority-weighted penalties to unfilled crew slots. Table 4.9 summarises the implementation.

The four changes address different aspects of the diagnosis. Additional technicians expand the available workforce, while overtime increases the daily capacity of eligible staff. Multi-day reservations allow long jobs to enter the plan, and carryover allows unassigned work to be reconsidered later. Priority penalties influence which staffing gaps the objective seeks to avoid; they do not create additional capacity.

With the combined configuration, assignment coverage rises from 198 jobs to 904 under static duration inputs and 851 under Random Forest predictions. As defined in Section 5.2, coverage counts jobs receiving at least one technician, including partially staffed

Table 4.9: Four-lever configuration used in the saved planning experiments.

Lever	Implemented change	Modelling scope and considerations
Workforce	Add 15 technicians across the skill pools and operating countries.	Expands eligible capacity; the added technicians also use updated travel-generation parameters.
Overtime	Set daily capacity to 600 minutes for overtime-eligible technicians.	Extends daily shift limits; uses a fixed 600-minute cap that interacts with capacity-reduction scenarios.
Multi-day logic	Truncate first-day capacity use, block subsequent whole days, and retry unassigned jobs up to the 30-day horizon limit.	Enables multi-shift repairs via external whole-day reservations; partially staffed two-person jobs are not retried.
Priority penalty	Penalise each unfilled crew slot using priority weights of 3, 2, and 1.	Prioritises critical and high-urgency coverage as a soft objective incentive rather than a hard constraint.

Source: own elaboration from the prototype scheduling rules.

two-technician jobs. These figures should therefore not be interpreted as counts of fully staffed and completed interventions.

The comparison demonstrates the effect of the combined configuration in this scenario. It does not identify the contribution of each lever separately, because all four are introduced together and the expanded workforce also uses updated travel parameters. Isolating their individual effects would require additional controlled experiments varying the interventions separately and in combination.

For decision support, the value of this exercise is the connection between a diagnosed restriction and a testable response. The initial result points to a capacity and modelling problem, while the revised configuration shows how changes to resources and planning rules alter assignment coverage. The detailed comparisons of cost, travel, coverage, and other planning indicators follow in Chapter 5.

4.7 From Diagnosis to Planning Decisions

The findings in this chapter provide a basis for selecting planning experiments. Regional imbalances suggest where additional qualified capacity should be considered. Duration bias motivates alternative estimates, while travel and skill diagnostics help explain why nominal capacity may not translate into feasible assignments. Historical lateness patterns identify further pressures to examine through disruption scenarios.

Table 4.10 connects each diagnostic finding with the evidence supporting it and the planning question it motivates. The purpose is to make the link between analysis and experimentation explicit.

For a field-service provider, this diagnostic process can help structure decisions about

Table 4.10: Diagnostic findings and the planning questions they motivate.

Diagnostic finding	Analytical evidence and scope	Decision-support question
Uneven resident capacity	Country demand-to-capacity ratios highlight regional imbalances prior to accounting for skills, travel, and crew multipliers.	Where should additional qualified capacity or cross-border allocation be tested?
Daily demand peaks	Baseline service demand exceeds standard-shift capacity on 15 days of the horizon.	Can subsequent spare capacity absorb deferred work through rolling carryover?
Systematic duration bias	Historical actual durations exceed static expected durations by 1.82 hours (14.8%) on average.	How do calibrated static and learned ML duration inputs reshape the generated plan?
Subgroup error variation	Out-of-fold Random Forest MAE varies across individual machine categories and operating countries.	Which operational segments merit targeted feature refinement or conditional buffers?
Recorded delay causes	Historical lateness labels capture ex-post delay attributions alongside multivariate complexity and SLA effects.	Which disruption assumptions (parts, travel, capacity) should be stress-tested?
Skill bottlenecks	Robotic-arm repairs exhibit the highest baseline ratio of jobs per qualified technician (12.1).	How does expanding certified skill pools affect overall assignment coverage?

Source: own elaboration from the diagnostic findings and prototype design.

staffing, skills, duration estimates, and working capacity. The prototype makes it possible to explore selected alternatives before drawing conclusions from aggregate ratios alone. However, the matrix also includes questions for further investigation, such as conditional prediction buffers and isolated skill additions, whose benefits are not established by the current experiments.

The following chapter evaluates two main sets of comparisons. The duration ablation compares raw static, calibrated static, Random Forest, and buffered estimates. The disruption experiments vary workforce availability, travel times, parts availability, and shift capacity. Together, they examine how the identified planning pressures affect assignment coverage and the other reported indicators within the synthetic scenario.

Chapter 5

Results and Discussion

This chapter evaluates the predictive and scheduling components of the prototype. The predictive evaluation uses the 3,000 historical assignments to compare duration estimates against recorded outcomes. The scheduling evaluation uses the 1,000 simulation jobs to examine how alternative duration inputs and operational disruptions affect the generated plans.

The analysis begins with the duration-model benchmark and the separate calibrated-static comparison. It then examines assignment coverage, staffing completeness, cost, travel, and the other planning indicators under five duration regimes. Finally, the disruption experiments test the response to changes in workforce availability, travel, parts, and daily capacity. The discussion connects these findings to the research questions and their practical interpretation.

5.1 Duration-Prediction Results

The predictive results address whether historical job and technician information improves duration estimation relative to the original static planning values. They also examine whether the implemented Random Forest offers an advantage over simpler alternatives.

5.1.1 Model Comparison and Prediction Errors

The original benchmark compares the static baseline, linear regression, Random Forest, and gradient boosting using shuffled 5-fold cross-validation across 3,000 historical assignments. Table 5.1 reports mean fold-level RMSE, MAE, and R^2, together with bias and residual standard deviation calculated from pooled out-of-fold predictions.

Random Forest improves on the raw static baseline, increasing mean fold-level R^2 from 0.694 to 0.748 and reducing RMSE from 4.823 to 4.377 hours and MAE from 3.154 to 2.998 hours. Its mean signed error is close to zero, compared with the static baseline's systematic underestimation of 1.820 hours.

Table 5.1: Predictive benchmark: 5-fold cross-validated error for four duration estimators ($N = 3{,}000$). Bias is the mean signed residual (predicted − actual).

Estimator	RMSE (h)	MAE (h)	R^2	Bias (h)	Resid. SD (h)
Static baseline	4.823	3.154	0.694	-1.820	4.468
Linear regression	4.278	2.932	0.759	-0.002	4.279
Random forest	4.377	2.998	0.748	$+0.004$	4.378
Gradient boosting	4.450	3.016	0.739	$+0.010$	4.451

Source: own elaboration from the saved predictive benchmark.

However, Random Forest does not produce the strongest aggregate point scores. Linear regression, supplied with the same engineered interaction features, achieves an R^2 of 0.759, an RMSE of 4.278 hours, and an MAE of 2.932 hours. All three learned models largely remove the downward bias of the original planning estimate. The evidence therefore supports correcting the static estimate, but does not establish that a tree ensemble is necessary to obtain that improvement.

Table 5.2: Variability of predictive performance: fold-level R^2 (mean $\pm$ standard deviation over five folds) and bootstrap 95% confidence intervals (200 resamples of the out-of-fold predictions).

Estimator	Fold R^2 (mean $\pm$ SD)	R^2 95% CI
Static baseline	0.694 ± 0.016	[0.679, 0.710]
Linear regression	0.759 ± 0.016	[0.745, 0.774]
Random forest	0.748 ± 0.016	[0.730, 0.763]
Gradient boosting	0.739 ± 0.018	[0.720, 0.756]

Source: own elaboration from the saved fold-level benchmark results.

Table 5.2 reports variability across folds and confidence intervals obtained from 200 bootstrap resamples of the fixed out-of-fold predictions. These intervals describe sampling variation in the retained predictions. They do not include the variability that would arise from repeatedly generating new populations and refitting the models, and they are not a formal paired test of model equivalence.

The residual distributions in Figure 5.1 illustrate the distinction between average bias and individual error. Static residuals are centred around -1.82 hours, while Random Forest residuals have a mean of $+0.004$ hours and a standard deviation of 4.378 hours. Despite its near-zero average bias, Random Forest still has an MAE of 2.998 hours against a mean actual duration of 14.12 hours. Individual interventions therefore remain difficult to predict precisely.

The fitted ensemble assigns 86.7% of its total impurity-based feature importance to `expected_duration_hours`. This indicates that the original planning estimate remains the main reference for prediction, with the other features supplying contextual adjustments. Feature importance describes the fitted trees and can be influenced by correlated inputs; it does not measure causal effects.

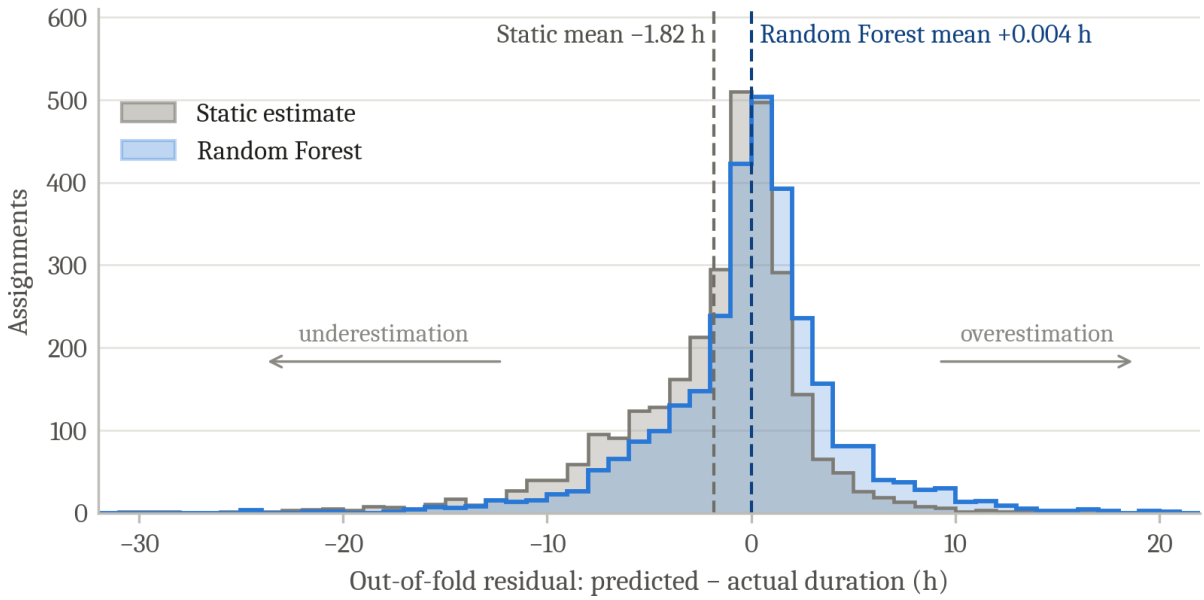


Figure 5.1: Out-of-fold residual distributions (predicted − actual duration). The static baseline is centred at -1.82 hours, evidencing systematic underestimation; the Random Forest is centred at zero.

Source: own elaboration from the saved out-of-fold duration predictions.

This reliance on the static estimate motivates a simpler question: how much improvement could be obtained by correcting that estimate with a single multiplier? The following comparison addresses this directly.

5.1.2 Comparison with Calibrated Static Estimates

A separate retained evaluation compares Random Forest with a calibrated static estimator on the same 3,000 historical assignments. It uses five shuffled folds with seed 42, after fixing the input order by job identifier and then technician identifier. Within each training fold, the calibration factor is calculated as the ratio of mean actual duration to mean expected duration. The factor is then applied to the held-out records.

Predictions are saved with their job, technician, and fold identifiers. This allows the two estimators to be compared on the same held-out assignments and supports the subgroup analysis in Table 4.4. No scheduling outputs from the associated later software version are used in the scheduling comparisons below.

In this evaluation, calibrated static estimates achieve a pooled RMSE of 4.337 hours, an MAE of 3.002 hours, and an R^2 of 0.753. Random Forest achieves a pooled RMSE of 4.365 hours, an MAE of 2.988 hours, and an R^2 of 0.750. The calibrated estimator has slightly better RMSE and R^2, while Random Forest has slightly better MAE.

These values must remain distinct from those in the original benchmark. Changing the input order changes membership of the shuffled folds, even when the random seed is unchanged. In the original run, the pooled Random Forest scores are an RMSE of 4.378 hours, an MAE of 2.998 hours, and an R^2 of 0.748. The separate job-identified evaluation is there-

fore not simply another presentation of the same predictions. The scheduling experiments retain the original fitted predictor and configuration.

For RQ1, the conclusion is that Random Forest improves on the raw planning estimate, particularly by removing systematic underestimation. However, a calibrated multiplier achieves similar aggregate accuracy in the separate evaluation, and linear regression has the strongest point scores in the original benchmark. The results do not establish Random Forest as the best estimator for this synthetic dataset.

5.2 Scheduling Results under Alternative Duration Estimates

The scheduling experiments examine what changes when alternative duration estimates are passed to the same planning formulation. The workforce, travel matrix, operational levers, solver settings, and candidate jobs are held fixed within the ablation. The reported differences therefore describe the plans generated under each duration regime.

5.2.1 Interpretation of Planning Indicators

The indicators require careful definition because the prototype represents daily assignments and coarse reservations rather than complete chronological execution.

- **Jobs with an assignment**: the number of unique simulation jobs receiving at least one technician during the 30-day horizon. Partially staffed two-technician jobs are included, so this measures assignment coverage rather than verified full-crew completion.
- **Unassigned jobs**: jobs receiving no technician assignment during the horizon. Jobs with an incomplete crew are not included in this count.
- **SLA proxy**: the number of assignment rows for which one-way base-to-job travel plus the service-duration input exceeds the contracted SLA window, expressed as a percentage of the 1,000 simulation jobs. It excludes accumulated carryover waiting, shift boundaries, and joint-crew arrival times. It is therefore a duration-exceedance indicator, not a calendar-based lateness rate.
- **Utilisation proxy**: the mean capped workload ratio over technician-days on which assignments start. Full travel and service minutes are divided by standard or scenario-reduced daily minutes and capped at 100%. The denominator is not the effective overtime cap. Inactive days and subsequent reserved days of multi-day jobs are excluded.

- **Crew fulfilment**: the proportion of two-technician jobs receiving at least two assignment rows. This measures staffing completeness without establishing synchronised arrival or execution.
- **Cost and risk**: reported cost sums post-processed technician-day route costs. Mean risk is the unweighted mean of route-level scores, with each route score averaging its assignment scores. It is not an assignment-weighted mean or an estimated probability.

These definitions are used consistently in the comparisons below. In particular, cost, SLA, and utilisation indicators need to be interpreted alongside assignment coverage and crew fulfilment.

5.2.2 Static and ML-Based Scheduling

Figure 5.2 compares selected indicators from the saved static, calibrated-static, and Random Forest plans. Table 5.3 presents the detailed comparison between the raw static baseline and ML mode.

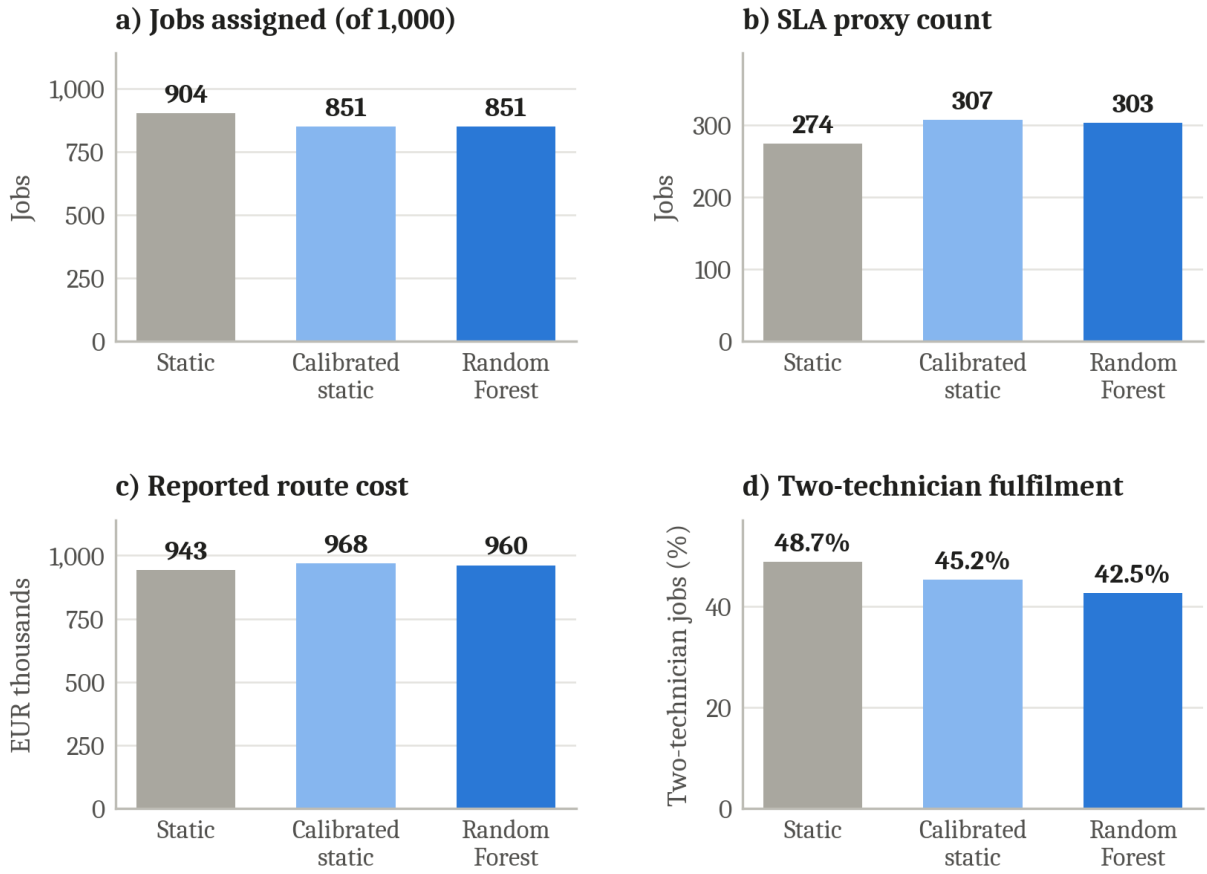


Figure 5.2: Selected KPIs from the saved 30-day static, calibrated-static and Random Forest plans. Each bar represents one retained deterministic run; no sampling interval is implied.

Source: own elaboration from the saved 30-day scheduling outputs.

Table 5.3: Baseline-vs-ML ablation: KPI comparison across the 30-day horizon.

KPI	Baseline	ML mode	Delta (ML − Baseline)
Jobs with an assignment	904 (90.4%)	851 (85.1%)	-53 (-5.3 pp)
Total cost	EUR 942,745	EUR 960,477	+EUR 17,732
Travel time	2,100 h 32 m	1,968 h 58 m	-131 h 34 m
SLA proxy count	274 (27.4%)	303 (30.3%)	$+29$ ($+2.9$ pp)
Utilisation proxy	96.8%	97.6%	$+0.8$ pp
Mean risk index	0.4365	0.4499	$+0.0134$
Two-technician fulfilment	48.7%	42.5%	-6.2 pp
Critical with assignment	100%	100%	0 pp
High with assignment	100%	100%	0 pp

Source: own elaboration from the saved static and Random Forest scheduling outputs.

The static baseline produces 1,015 assignment rows covering 904 unique jobs. ML mode produces 948 assignment rows covering 851 jobs, a reduction of 53 jobs or 5.3 percentage points. Within the baseline's 904-job coverage count, 35 two-technician jobs receive only one technician. Overall two-technician fulfilment falls from 48.7% under static inputs to 42.5% under Random Forest inputs.

These results show why prediction accuracy and assignment coverage need separate interpretation. The raw static estimates systematically understate historical service duration. Correcting that bias increases the capacity required by the modelled workload and can reduce the number of assignments accommodated within the same workforce configuration. Higher coverage under the raw baseline is therefore not sufficient evidence of a better executable schedule.

Both regimes assign at least one technician to every critical and high-priority job. The reduction in unique-job coverage consequently falls on normal-priority demand. However, the crew-fulfilment results show that initial coverage of urgent work does not establish complete staffing of every urgent two-technician job.

ML mode reduces total travel by 131 hours 34 minutes, approximately 6.3%, while increasing reported cost by EUR 17,732. The higher reported cost, despite fewer assignment rows, is consistent with the accounting of selected technician-day routes using estimated service and travel hours and overtime rates where applicable. The mean reported cost per assignment row rises from about EUR 929 in the static plan to about EUR 1,013 in the Random Forest plan. The lower static total should be read alongside its shorter duration inputs, which the historical data show to be biased low. The travel reduction accompanies fewer assignment rows. Mean travel per assignment is approximately 2.069 hours in the static plan and 2.077 hours in the Random Forest plan, so lower aggregate travel does not demonstrate shorter average journeys.

Reported cost also depends on the selected assignments, estimated service hours, and post-solver accounting rules. Similarly, the SLA-proxy increase from 274 to 303 is calcu-

lated using different duration inputs in the two plans. It cannot be interpreted as an observed increase in realised lateness.

5.2.3 Calibrated Estimates and Safety Buffers

The extended ablation (Table 5.4) compares five duration regimes: raw static estimates, calibrated static estimates, Random Forest point predictions, and Random Forest predictions with two global safety buffers. The planning calibration factor is 1.148, and the residual standard deviation used for buffering is $\sigma = 4.38$ hours.

These planning parameters are derived from historical data. The simulation jobs contain no realised duration targets. This differs from predictive validation, where the calibration factor is fitted within each training fold before evaluating held-out records.

Table 5.4: Duration-estimate ablation over the 30-day horizon, optimiser and four levers held fixed. Jobs with assignment $= 1{,}000-$ unassigned. $\sigma = 4.38$ h is the out-of-fold residual standard deviation.

Duration regime	Jobs with assignment	Cost (EUR)	SLA proxy	Util. (%)
Static (raw)	904	942,745	274 (27.4%)	96.8
Calibrated static ($\times 1.148$)	851	968,392	307 (30.7%)	97.6
Random forest (mean)	851	960,477	303 (30.3%)	97.6
Random forest ($+0.5\sigma$)	775	983,817	338 (33.8%)	99.5
Random forest ($+\sigma$)	700	1,012,707	368 (36.8%)	100.0

Source: own elaboration from the saved duration-estimate ablation.

Calibrated static estimates and unbuffered Random Forest predictions both produce coverage of 851 jobs and a utilisation proxy rounding to 97.6%. This similarity is consistent with an important role for correcting average duration, although equal coverage does not mean that the selected jobs and technician assignments are identical.

Among these two plans, Random Forest produces lower reported cost, EUR 960,477 compared with EUR 968,392, and fewer SLA-proxy exceedances, 303 compared with 307. It also requires less aggregate travel, 1,968 h 58 m compared with 2,007 h 31 m. These differences describe the plans under their respective duration assumptions. They do not independently establish better performance under common realised execution times.

Adding 0.5σ and 1.0σ buffers reduces coverage to 775 and 700 jobs. The utilisation proxy rises to 99.5% and 100.0%, respectively. Longer planning allowances leave less capacity for additional assignments and increase the number of duration-based SLA exceedances.

The buffered regimes therefore demonstrate the planning cost of applying a uniform safety margin. They do not establish whether the resulting plans would be more reliable during execution. That assessment would require realised durations or a common execution simulator. The results motivate investigating calibrated, job-specific uncertainty estimates rather than assuming that every job needs the same additional allowance.

5.2.4 Solver Performance and Computational Time

Table 5.5 reports the computational profile of the 30 daily CP-SAT subproblems in static and ML modes. Both regimes achieve proven optimality with a 0.00% gap on all 30 days. These certificates apply to each day's encoded problem, conditional on the reservations and decisions already made.

Table 5.5: Solver computational profile over the 30 daily CP-SAT subproblems, both duration modes (65 technicians, 30-day horizon).

Metric	Baseline	ML mode
Days proven optimal	30 / 30	30 / 30
Feasible-only / failed	0 / 0	0 / 0
Maximum optimality gap	0.00%	0.00%
Total solve time (s)	1.19	1.73
Solve time per day, min/mean/max (s)	0.017 / 0.040 / 0.070	0.014 / 0.058 / 0.119
Variables per day, min/mean/max	1,344 / 3,004 / 4,826	1,075 / 3,736 / 6,688
Constraints per day, min/mean/max	791 / 1,835 / 2,985	640 / 2,239 / 3,939

Source: own elaboration from the saved daily solver statistics.

The variable and constraint counts describe the scale of the daily models. They are calculated by the implementation from the technician, job, and eligibility structure; they are not presolve statistics extracted from CP-SAT.

Total solver time is 1.19 seconds for the static baseline and 1.73 seconds for ML mode. Recorded end-to-end application time is 2.19 seconds and 271.19 seconds, respectively. The latter includes feature construction, repeated candidate-pair prediction calls, model construction, and post-processing. Since no component-level timing breakdown was recorded, the additional time cannot be attributed quantitatively to prediction alone.

These results show that the retained daily assignment problems are solved quickly. They do not establish full-month optimality, because the daily decomposition does not reconsider earlier commitments jointly. They also show why solver runtime and user-facing application runtime need to be reported separately.

5.3 Response to Operational Disruptions

The disruption experiments examine how assignment coverage and other planning indicators respond to changes in operational inputs. The comparisons in this section use the static-mode baseline, which has 96 unassigned jobs and a reported cost of EUR 942,745. Individual changes are examined first, followed by combined disruptions.

5.3.1 Single-Factor Sensitivity

Figure 5.3 and Table 5.6 report one-factor-at-a-time changes in workforce availability, travel time, and nominal shift duration. Other settings remain at the reference configura-

tion within each comparison.

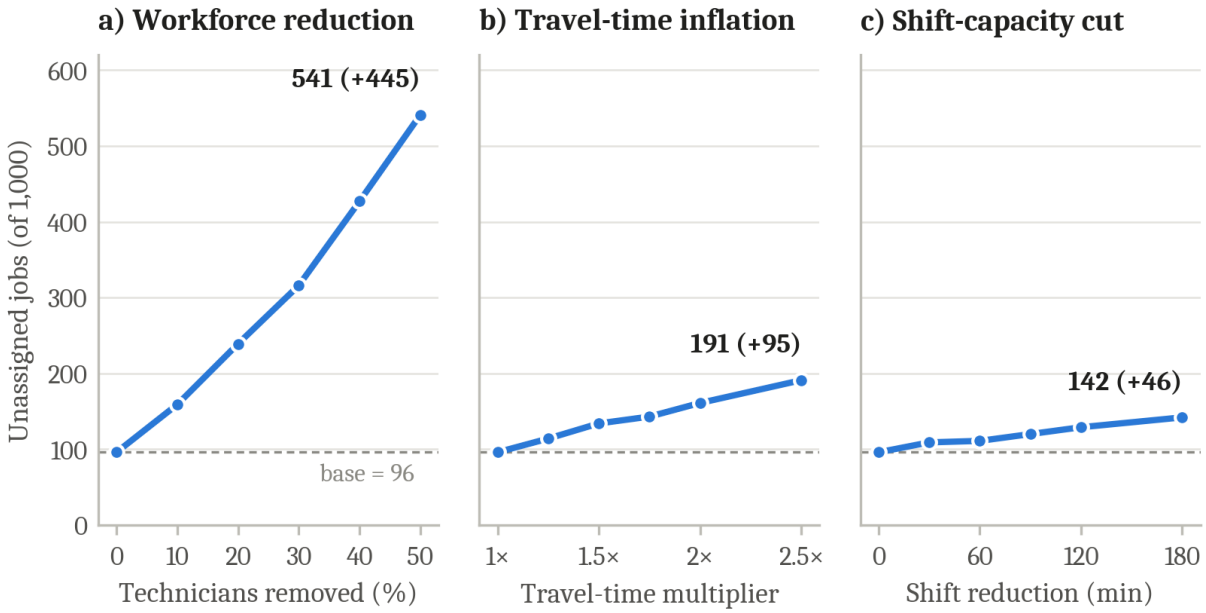


Figure 5.3: Single-factor sensitivity of unassigned jobs over the 30-day horizon. Archived static-mode sensitivity outputs. Interpretation of capacity cuts is limited by the fixed overtime cap.

Source: own elaboration from the archived sensitivity-analysis outputs.

Table 5.6: Single-factor sensitivity: unassigned jobs as each operational lever is varied about the base configuration (30-day horizon).

Workforce cut	Unassign.	Travel mult.	Unassign.	Shift cut	Unassign.
0%	96	1.00×	96	−0 min	96
10%	159	1.25×	114	−30 min	109
20%	239	1.50×	134	−60 min	111
30%	316	1.75×	143	−90 min	120
40%	427	2.00×	161	−120 min	129
50%	541	2.50×	191	−180 min	142

Source: own elaboration from the archived sensitivity-analysis outputs.

Within the tested ranges, workforce reductions produce the largest deterioration in coverage. Removing 10% of technicians increases unassigned jobs from 96 to 159, while removing 50% increases them to 541. This response is consistent with the limited spare capacity and specialised skill pools identified in the operational diagnosis.

Longer travel times also reduce coverage. Doubling the travel-time inputs raises unassigned work from 96 to 161 jobs. In the model, these longer journeys consume capacity that would otherwise be available for service work and can affect multi-day reservations.

Nominal shift cuts produce a smaller response, with a 180-minute reduction increasing unassigned jobs from 96 to 142. This result depends on the implemented overtime rule: eligible technicians retain a fixed 600-minute effective capacity when overtime is enabled. The experiment therefore does not represent an equal reduction in working capac-

ity across the entire workforce.

Parts disruptions behave differently in static mode. Changing parts availability affects the post-solver risk score, but does not change the static duration input or act as a hard assignment restriction. Static assignment counts consequently remain unchanged. This outcome reflects the model's treatment of parts, rather than evidence that stockouts would have no operational effect.

5.3.2 Combined Disruption Scenarios

The detailed compound scenario combines a 20% workforce reduction, a $1.6\times$ travel-time multiplier, a 10% parts-disruption sampling parameter, and a 90-minute nominal daily capacity cut. Table 5.7 compares its planning indicators with the undisrupted static baseline.

Table 5.7: Compound stress scenario: baseline-vs-scenario KPI comparison. Risk values reproduce the rounded dashboard display; the saved baseline value is 0.4365. Travel deltas here are calculated from the displayed hour-and-minute totals.

KPI	Baseline	Scenario	Delta
Total cost	EUR 942,745	EUR 745,169	−EUR 197,576
Travel time	2,100 h 32 m	1,727 h 50 m	-372 h 42 m
SLA proxy count	274 (27.4%)	276 (27.6%)	$+2$ ($+0.2$ pp)
Unassigned jobs	96	260	$+164$
Utilisation proxy	96.8%	98.5%	$+1.7$ pp
Mean risk index (displayed)	0.436	0.456	$+0.020$
Two-technician fulfilment	48.7%	38.6%	-10.1 pp

Source: own elaboration from the compound-scenario results reported in this chapter.

Unassigned jobs increase from 96 to 260, while two-technician fulfilment falls from 48.7% to 38.6%. Reported cost decreases by EUR 197,576 and total travel falls by 372 hours 42 minutes. These reductions accompany a substantial decline in assigned work and should not be read as efficiency gains.

The SLA proxy increases by only two cases despite the deterioration in coverage. This illustrates a limitation of the indicator: jobs receiving no assignment do not contribute an assignment-row duration exceedance. Their accumulated waiting is also not represented in the proxy. The unassigned-job count and crew-fulfilment measure are therefore essential for interpreting the scenario.

Five named stress tests provide further examples of combined operating conditions, as shown in Table 5.8. Parts percentages specify the share of jobs sampled for possible stockout, not the final proportion of jobs lacking required parts.

Staffing Crisis and Perfect Storm produce the largest increases in unassigned work among these named scenarios, at 220 and 190 additional jobs. Their lower reported expenditure again accompanies fewer assignments. Winter Storm increases unassigned work by 44 jobs under combined travel and nominal shift changes. Regulatory Shift Cut adds 33

Table 5.8: Multi-factor stress tests over the 30-day horizon. Each delta is the change relative to the undisrupted baseline. Parts percentages denote the share of jobs sampled for possible stockout, not the final stockout rate.

Scenario	Composition	Δ Unassign.	Δ SLA	Δ Cost (EUR)
Winter Storm	$1.5\times$ travel, 25% parts disruption, -30 min shifts	$+44$	$+13$	$+886$
Staffing Crisis	30% workforce cut, 20% parts disruption	$+220$	$+3$	$-259{,}065$
Supply Chain Failure	50% parts disruption	0	0	0
Regulatory Shift Cut	-120 min daily shift capacity	$+33$	-1	$-4{,}931$
Perfect Storm	20% workforce cut, $1.4\times$ travel, 30% parts disruption, -30 min	$+190$	$+19$	$-191{,}754$

Source: own elaboration from the archived stress-test outputs.

unassigned jobs, with its effect moderated by the fixed overtime capacity. Because each scenario is reoptimised through sequential daily windows, assignment counts need not be monotonic in the nominal travel and shift changes, even when the same fixed-seed 20% workforce sample is removed.

Supply Chain Failure leaves the static assignment plan and the listed cost and SLA indicators unchanged because parts availability does not restrict static scheduling feasibility. Its effect is visible through the risk-scoring mechanism rather than through the assignment counts in this table.

For RQ4, these experiments show how the prototype exposes sensitivity to specified operational changes. They also demonstrate that scenario interpretation requires an understanding of the implemented rules: an unchanged indicator may reflect a missing dependency in the model rather than resilience in a real service operation.

5.4 Discussion and Planning Implications

The results connect duration prediction, capacity diagnosis, and decision support, but they do not point to a simple conclusion that adding Machine Learning always improves every scheduling indicator.

First, much of the predictive improvement comes from correcting systematic duration bias. Random Forest improves on the raw static estimate, but linear regression and calibrated static estimates remain competitive. For the scenario studied here, the value of revising the original planning assumptions is clearer than the value of increasing model complexity. This supports evaluating simple comparators before attributing improvements to a particular algorithm.

Second, improved duration estimates can reveal a less favourable capacity position.

Raw static inputs produce 904 covered jobs, while calibrated static and Random Forest inputs produce 851. This reduction is consistent with longer duration allowances consuming more of the available capacity. A dispatcher using these results would need to distinguish a plan that appears to cover more work from one whose duration assumptions are better supported by the historical data.

The present experiments cannot determine which plan would complete more jobs in practice, because the simulation jobs have no common realised execution outcomes. They establish sensitivity of the generated plans to duration assumptions. A stronger comparison would execute the competing plans against the same job–technician outcomes and measure completion, waiting, incurred cost, and lateness.

Third, aggregate cost and travel must be read alongside the amount of work assigned. The disruption scenarios show that reported expenditure can fall while unassigned demand rises sharply. Similarly, lower aggregate travel in ML mode does not imply shorter average journeys. Comparing these totals without coverage and crew completeness could therefore produce a misleading managerial interpretation.

A duration correction should also not be applied mechanically to total cost. Multiplying baseline expenditure by 14.8% would adjust travel fares as well as labour, while ignoring changes in the selected assignments. The ablation instead reruns the scheduling model under each duration regime. Its cost outputs remain subject to the accounting conventions of the prototype.

Fourth, the capacity experiments show that the scheduling representation itself is consequential. Multi-day reservations allow long repairs to enter the plan, but partial crews, whole-day blocking, and fixed overtime caps shape the resulting indicators. Daily optimality certificates confirm the solution quality of the encoded subproblems; they do not validate these modelling assumptions or establish full-horizon optimality.

The decision-support contribution is therefore the ability to inspect these relationships together. The prototype links duration estimates to assignments, exposes unallocated demand, and allows selected operating assumptions to be varied. It helps formulate and compare planning alternatives within a controlled scenario. Moving from those comparisons to operational recommendations requires the additional validation and implementation work discussed in the next chapter.

Chapter 6

Conclusions, Limitations and Future Work

This thesis developed and evaluated a decision-support prototype for field technician scheduling in a synthetic European industrial service setting. The system combines duration prediction, daily CP-SAT assignment, multi-day reservations, heuristic risk scoring, and an interactive interface for inspecting plans and scenarios. This chapter summarises the answers to the research questions, identifies the boundaries of the evidence, and sets out priorities for further development.

6.1 Conclusions by Research Question

RQ1: Duration prediction. Random Forest improves on the raw static estimate in the historical benchmark, increasing mean fold-level R^2 from 0.694 to 0.748 and reducing RMSE from 4.823 to 4.377 hours and MAE from 3.154 to 2.998 hours. Its mean signed error is approximately zero, compared with the static baseline's -1.820-hour bias. However, linear regression achieves slightly better aggregate point scores, with $R^2 = 0.759$ and MAE of 2.932 hours. In the separate calibrated-static evaluation, a single multiplier achieves a pooled R^2 of 0.753 compared with 0.750 for Random Forest. The evidence therefore supports correcting duration bias, while providing no clear basis for claiming that Random Forest is the best available estimator in this synthetic setting.

RQ2: Scheduling outcomes. Alternative duration inputs change the assignments produced by the same scheduling model. Coverage falls from 904 jobs under raw static inputs to 851 under both calibrated static and Random Forest inputs. The static and Random Forest regimes retain at least one assignment for all critical and high-priority jobs, although complete two-technician staffing is not guaranteed. Between the two 851-job plans, Random Forest has lower reported cost, EUR 960,477 compared with EUR 968,392, and lower aggregate travel, 1,968 h 58 m compared with 2,007 h 31 m. Adding 0.5σ and 1.0σ safety buffers reduces coverage further to 775 and 700 jobs. These results establish planning

sensitivity to duration assumptions, rather than realised savings or completion improvements.

RQ3: Capacity diagnosis. The diagnostics identify the single-shift restriction, regional workload imbalance, and specialised skill availability as important constraints in the scenario. Italy has the greatest regional service-hour pressure, while robotic-arm and PLC qualifications have particularly high demand relative to eligible headcount. The combined intervention of 15 additional technicians, selective overtime, multi-day reservations with carryover, and priority-weighted shortfall penalties raises static-mode assignment coverage from 198 to 904 jobs. Because these changes were introduced together, alongside updated travel parameters for the additional technicians, the comparison evaluates the combined configuration rather than the isolated effect of each intervention.

RQ4: Scenario response. Within the tested ranges, workforce reductions cause the largest losses in assignment coverage, while longer travel times also increase unassigned work. The experiments reveal the influence of specific implementation rules: parts disruptions change the risk score without changing static assignment counts, and nominal shift cuts are partly cushioned by the fixed 600-minute capacity of overtime-eligible technicians. Scenario analysis therefore provides information both about capacity pressure and about how the model represents operational dependencies.

Taken together, the findings show how duration estimation and constrained assignment can be combined in an inspectable planning prototype. Its contribution is the implementation and evaluation of this relationship within a defined scenario, including the diagnosis of capacity restrictions and the comparison of alternative assumptions. The results support further development and testing rather than immediate deployment as a validated operational scheduling system.

6.2 Limitations

The findings depend on the synthetic data, the scheduling representation, and the implementation used for the retained experiments. The following limitations define how far the results can be interpreted.

6.2.1 Data and Predictive Validation

The 3,000 historical assignments and 1,000 simulation jobs are separate datasets, but both originate from the same synthetic generator. Learned relationships may therefore reflect the generation rules rather than patterns that would persist in enterprise records. The experiments do not demonstrate cost savings or service improvements in an operating company.

The shuffled cross-validation design allows the same technicians to appear in training and validation folds. It evaluates held-out assignments within the synthetic historical

population, rather than generalisation to entirely new technicians or later operating conditions. Keeping simulation jobs outside model training does not remove this limitation.

The simulation jobs also have no realised execution-duration targets. Consequently, competing schedules cannot be scored against common future outcomes. Predictive fit and planning indicators are evaluated separately, and improved historical accuracy cannot be assumed to imply improved realised scheduling performance.

The model provides point estimates, while the buffered regimes apply uniform residual-based offsets. These are not calibrated job-specific prediction intervals. Impurity-based feature importance similarly describes how the ensemble uses its inputs rather than identifying causal effects or the impact of a feature on scheduling decisions.

Subgroup MAE comparisons and historical lateness summaries are descriptive. Joint subgroup effects, such as the combination of priority, machine category, and country, would require dedicated analysis with sufficient observations. The bootstrap intervals in Table 5.2 use fixed out-of-fold predictions and do not include model-refitting variability across new populations. Formal paired comparisons would be needed to establish model differences or equivalence more rigorously.

6.2.2 Scheduling Representation and Planning Indicators

Travel is represented as an independent one-way base-to-job leg for each assignment. Return journeys and travel between successive jobs are excluded from the optimisation model. The displayed visit sequences do not change this representation.

The horizon is solved through sequential daily subproblems. Optimality on all 30 days applies only to those subproblems, conditional on earlier commitments. It does not establish that the resulting monthly plan is globally optimal.

Multi-day jobs consume capped capacity on their start day and reserve subsequent whole days outside the solver. This does not represent precise execution times, unused capacity on a final partial day, or guaranteed completion before the end of the planning horizon.

Two-technician jobs can receive only one technician and still count as covered. They are not re-queued to fill the missing crew slot, and synchronised arrival is not enforced. Assignment coverage therefore differs from complete staffing and completed service.

The SLA proxy evaluates individual travel-plus-service durations without accumulating waiting from carryover or reconstructing completion times. The utilisation proxy includes assignment-start days and caps workload ratios, while excluding inactive days and later reserved days. These measures describe particular aspects of the generated plans rather than complete operational lateness or workforce utilisation.

The geographic and skill diagnostics also simplify the assignment problem. Country capacity ratios, nearest-base distances, and jobs-per-qualified-technician ratios do not jointly account for travel, overlapping qualifications, and crew requirements. They identify

planning pressures but do not establish feasible allocations or financial returns.

6.2.3 Scenario Comparisons and Cost Accounting

The four capacity interventions are introduced jointly in the retained experiments. Their separate contributions and interactions are not isolated. In addition, the 15 added technicians use different travel-generation parameters from the initial workforce, so the comparison includes changes in travel inputs as well as headcount.

Coordinate-based geographic diagnostics and the stored travel matrix use different distance representations. The map-based findings therefore cannot be treated as a direct reconstruction of the travel inputs used by the scheduling model.

Scenario behaviour also depends on implementation choices. Overtime-eligible technicians retain a fixed 600-minute capacity when overtime is enabled, even when nominal shifts are reduced. Parts stockouts do not prevent assignments; they affect the risk score in static mode and may also affect duration predictions in ML mode. The parts-disruption percentage controls sampling rather than the final stockout prevalence.

Reported cost differs from the objective minimised by CP-SAT. The objective costs assignments separately after converting duration to integer minutes and charges overtime only when it is enabled and permitted for the technician. Reporting instead aggregates travel and service hours by technician-day route, applies overtime rates above the standard or reduced shift regardless of eligibility, and rounds route costs.

Neither accounting convention resets the regular-rate allowance on each reserved day of a multi-day job. The retained cost totals must therefore be interpreted as prototype accounting outputs, rather than validated estimates of incurred operational expenditure.

6.2.4 Prototype Validation and Reproducibility

The application was developed and demonstrated as a research prototype. It does not include production authentication, role-based access control, or formal user-acceptance testing with commercial dispatchers. Its usability and value in an operational workflow have therefore not been established through field trials.

Reproducibility also requires preserving the relationship between data versions, fitted models, solver configurations, and saved outputs. The original predictive benchmark and the later job-identified evaluation have different fold memberships and must remain separately documented. The scheduling experiments retain their original configuration; later software changes should not be used to reinterpret archived results without rerunning the relevant comparisons.

6.3 Future Work

The next stage should first strengthen the execution and evaluation rules before expanding the range of application features.

A first priority is a consistent benchmark in which the solver objective, reported cost, and completion-based SLA measures follow aligned accounting rules. Capacity-reduction scenarios should also reduce overtime capacity consistently with their intended interpretation. A strict all-or-nothing crew option with synchronised starts would allow comparison with the current partial-staffing policy.

A second priority is chronological execution evaluation. Competing schedules should be tested against common realised job–technician durations, either through operational records or a validated execution simulator. Static, calibrated-static, linear-regression, and Random Forest policies could then be compared using completed jobs, waiting, cascading delays, overtime, and incurred cost. A rolling perfect-information benchmark could help quantify the consequences of uncertain duration inputs.

Predictive validation should also be extended. Technician-grouped and chronological splits would assess performance on unseen staff and changing operating conditions. Evaluation on real field-service records would test whether the relationships learned from synthetic data transfer to practice. Conformal prediction or quantile regression forests could provide job-specific safety margins, subject to checking calibration on the population where they are used.

The scheduling approach could be assessed against a multi-day or full-horizon MIP or CP-SAT formulation on smaller instances. This would quantify the effect of daily decomposition and coarse reservations. Separate intervention experiments would help distinguish the effects of workforce expansion, overtime, multi-day logic, and priority penalties. Decision-aware learning, including Smart Predict-then-Optimise approaches adapted to duration-dependent feasibility, offers a further extension [10], [11].

Finally, industrial use would require consistent road and flight travel inputs, integration with enterprise work-order and inventory systems, authentication, audit logging, and structured evaluation with field-service planners. These developments would allow the prototype's planning functions to be tested within an actual service workflow and establish whether the diagnostic benefits observed in the synthetic scenario translate into better operational decisions.

List of Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
CP	Constraint Programming
CP-SAT	Constraint Programming – Satisfiability
DSS	Decision Support System
FSM	Field Service Management
FTSP	Field Technician Scheduling Problem
KPI	Key Performance Indicator
MAE	Mean Absolute Error
MIP	Mixed-Integer Programming
ML	Machine Learning
OR	Operations Research
PLC	Programmable Logic Controller
RF	Random Forest
RMSE	Root Mean Squared Error
RQ	Research Question
SLA	Service Level Agreement
SPO	Smart Predict-then-Optimise
TRSP	Technician Routing and Scheduling Problem
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows

List of Figures

3.1 Input datasets and join relationships. 17
3.2 The two-phase pipeline. 25
3.3 The capacity bottleneck. 30

4.1 Geographic distribution of technician bases and job sites. 36
4.2 Planned and actual historical durations. 39
4.3 Recorded causes among the 632 late assignments. 40
4.4 Travel mode distribution (left) and travel duration distribution (right). . . . 43
4.5 Simulation job distributions. 44

5.1 Out-of-fold residual distributions. 51
5.2 Selected KPIs from the saved 30-day plans. 53
5.3 Single-factor sensitivity of unassigned jobs. 57

List of Screenshots

3.1 Overview: indicators for the saved plan. 22
3.2 Map: selected route and its popup. 22
3.3 Timeline: selected assignment and route detail. 23
3.4 Jobs: filtered records and selected job detail. 23
3.5 Optimisation and Analysis: planning controls. 24

List of Tables

3.1 Skill compatibility matrix (initial 50-technician workforce). 30
3.2 Composite risk score: components and weights. 32

4.1 Country-level coverage deficit. 37
4.2 Nearest eligible technician: coordinate distances. 37
4.3 Capacity ratios after one local technician is added. 38
4.4 Random Forest error by subgroup. 39
4.5 Cause of lateness cross-tabulated by priority. 40
4.6 Cause of lateness cross-tabulated by country. 41
4.7 Cause of lateness cross-tabulated by machine category. 41
4.8 Logistic regression of SLA lateness on operational drivers. 42
4.9 Four-lever configuration used in the saved planning experiments. 46
4.10 Diagnostic findings and the planning questions they motivate. 47

5.1 Predictive benchmark: 5-fold cross-validated error. 50
5.2 Variability of predictive performance. 50
5.3 Baseline-vs-ML ablation: KPI comparison across the 30-day horizon. 54
5.4 Duration-estimate ablation over the 30-day horizon. 55
5.5 Solver computational profile over the daily subproblems. 56
5.6 Single-factor sensitivity of unassigned jobs. 57
5.7 Compound stress scenario: KPI comparison. 58
5.8 Multi-factor stress tests over the 30-day horizon. 59

References

[1] J. A. Castillo-Salazar, D. Landa-Silva, and R. Qu, "Workforce scheduling and routing problems: literature survey and computational study," *Ann. Oper. Res.*, vol. 239, no. 1, pp. 39–67, 2016. doi: 10.1007/s10479-014-1687-2.

[2] G. D. Konstantakopoulos, S. P. Gayialis, and E. P. Kechagias, "Vehicle routing problem and related algorithms for logistics distribution: a literature review and classification," *Oper. Res.*, vol. 22, no. 3, pp. 2033–2062, 2022. doi: 10.1007/s12351-020-00600-7.

[3] J.-F. Cordeau, G. Laporte, F. Pasin, and S. Ropke, "Scheduling technicians and tasks in a telecommunications company," *J. Sched.*, vol. 13, no. 4, pp. 393–409, 2010. doi: 10.1007/s10951-010-0188-7.

[4] Global Market Insights, "Field Service Management Market Size, Forecasts Report 2035," *Industry analysis report*, Dec. 2025. [Online]. Available: https://www.gminsights.com/industry-analysis/field-service-management-market. [Accessed: Sept. 23, 2026].

[5] Mordor Intelligence, "Field Service Management Market Report: Industry Trends, Size and Growth Analysis 2031," *Industry analysis report*, Feb. 2026. [Online]. Available: https://www.mordorintelligence.com/industry-reports/field-service-management-market. [Accessed: Sept. 23, 2026].

[6] V. F. Yu, Y.-S. Lin, P. Jodiawan, S.-W. Lin, and Y.-C. Lai, "The field technician scheduling problem with experience-dependent service times," *Mathematics*, vol. 11, no. 21, art. 4562, 2023. doi: 10.3390/math11214562.

[7] V. Pillac, C. Guéret, and A. L. Medaglia, "A parallel matheuristic for the technician routing and scheduling problem," *Optim. Lett.*, vol. 7, no. 7, pp. 1525–1535, 2013. doi: 10.1007/s11590-012-0567-4.

[8] B. Toaza and D. Esztergár-Kiss, "A review of metaheuristic algorithms for solving TSP-based scheduling optimization problems," *Appl. Soft Comput.*, vol. 148, art. 110908, 2023. doi: 10.1016/j.asoc.2023.110908.

[9] M. Gamst and D. Pisinger, “Decision support for the technician routing and scheduling problem,” *Networks*, vol. 83, no. 1, pp. 169–196, 2024. doi: 10.1002/net.22188.

[10] A. N. Elmachtoub and P. Grigas, “Smart ‘Predict, then Optimize’,” *Manage. Sci.*, vol. 68, no. 1, pp. 9–26, 2022. doi: 10.1287/mnsc.2020.3922.

[11] J. Mandi, E. Demirović, P. J. Stuckey, and T. Guns, “Smart predict-and-optimize for hard combinatorial optimization problems,” in *Proc. AAAI Conf. Artificial Intelligence*, vol. 34, no. 2, 2020, pp. 1603–1610. doi: 10.1609/aaai.v34i02.5521.

[12] F. Yalaoui, T. Arbaoui, and Y. Ouazene, Eds., *Planning and Scheduling Optimization*. Basel, Switzerland: MDPI, 2021. doi: 10.3390/books978-3-0365-2338-5.

[13] Google, “CP-SAT Solver,” *OR-Tools documentation*, Aug. 2024. [Online]. Available: https://developers.google.com/optimization/cp/cp_solver. [Accessed: Sept. 23, 2026].

[14] L. Glomb, F. Liers, and F. Rösel, “A rolling-horizon approach for multi-period optimization,” *Eur. J. Oper. Res.*, vol. 300, no. 1, pp. 189–206, 2022. doi: 10.1016/j.ejor.2021.07.043.

[15] T. P. Carvalho, F. A. A. M. N. Soares, R. Vita, R. d. P. Francisco, J. P. Basto, and S. G. S. Alcala, “A systematic literature review of machine learning methods applied to predictive maintenance,” *Comput. Ind. Eng.*, vol. 137, art. 106024, 2019. doi: 10.1016/j.cie.2019.106024.

[16] D. Zuev, A. Kalistratov, and A. Zuev, “Machine learning in IT service management,” *Procedia Comput. Sci.*, vol. 145, pp. 675–679, 2018. doi: 10.1016/j.procs.2018.11.063.

[17] V. Poutiainen, “Utilizing artificial intelligence to enhance task duration estimation and optimize workforce qualifications in the field service industry,” M.Sc. thesis, Lappeenranta–Lahti Univ. of Technology (LUT), Finland, 2025.

[18] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001. doi: 10.1023/A:1010933404324.

[19] S. Gupta, S. Modgil, S. Bhattacharyya, and I. Bose, “Artificial intelligence for decision support systems in the field of operations research: review and future scope of research,” *Ann. Oper. Res.*, vol. 308, no. 1, pp. 215–274, 2022. doi: 10.1007/s10479-020-03856-6.

[20] I. A. Arin, A. A. Andika, A. S. Putra, and M. R. Febriano, “The development of a field service management information system to help companies manage field workers: proposed prototype,” in *Proc. 12th Int. Conf. Cyber and IT Service Management (CITSM)*, IEEE, 2024, pp. 1–6. doi: 10.1109/CITSM64103.2024.10775321.

[21] J. F. Gómez Fernández and A. Crespo Márquez, “Artificial intelligence and machine learning in industrial maintenance optimization,” in *Digital Maintenance and Asset Digitalization* (Engineering Asset Management Review, vol. 5), A. Crespo Márquez et al., Eds. Cham: Springer, 2026, pp. 119–143. doi: 10.1007/978-3-032-21502-4_6.

[22] H. Park, D. Son, B. Koo, and B. Jeong, “Waiting strategy for the vehicle routing problem with simultaneous pickup and delivery using genetic algorithm,” *Expert Syst. Appl.*, vol. 165, art. 113959, 2021. doi: 10.1016/j.eswa.2020.113959.