Demand-buffer robustification of deterministic staff scheduling under stochastic demand

Marta Inês Brandão Pereira

Master's Dissertation

Supervisors:
Prof. Gonçalo Figueira
Prof. Fábio Neves-Moreira

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Master in Industrial Engineering and Management

2026-06-25

Abstract

Staff scheduling is a central operational problem in service organisations, where the number of workers needed varies throughout the day and the week. Matching staffing to this varying demand is economically consequential: understaffing leaves demand uncovered and erodes service, while overstaffing wastes scarce labour. What makes the problem hard is that demand is uncertain, so a schedule fixed in advance can perform poorly once demand is realised.

Such schedules are commonly produced by deterministic Mixed-Integer Programming (MIP) models that optimise against a single nominal staffing requirement treated as certain. When realised demand deviates from that profile, particularly in its right tail, the resulting schedule can leave substantial coverage shortfall, not because the workforce is too small or the optimiser too weak, but because a single fixed schedule built on a single fixed requirement cannot absorb the variability around it.

This dissertation asks whether a deterministic MIP can be made more robust to demand uncertainty without being redesigned, by acting only on the requirement profile supplied to it. A demand buffer is proposed that raises the nominal profile using demand variability estimated from historical data, in two specifications: a global buffer with one conservatism level, calibrated by grid search, and a finer block-specific buffer, calibrated by Bayesian optimisation. Buffer parameters are tuned on training scenarios through a simulation-based layer and evaluated out of sample at two levels of workforce loading.

On a fixed workforce, the buffer reduces out-of-sample shortfall by 11.7% when the workforce is tightly loaded and 49.1% when it has slack, without adding capacity, by reallocating a constant coverage budget towards the more variable periods; tail risk falls by almost as much as the mean, so the gain carries no average-versus-tail trade-off. The finer block-specific buffer yields gains in some windows but no reliable improvement over the global one on average. Demand-buffer robustification is therefore an effective, low-cost way to make a deterministic scheduler more robust to stochastic demand, most valuable when the workforce is not tightly loaded.

Keywords: Staff Scheduling; Demand Uncertainty; Schedule Robustification; Mixed-Integer Programming (MIP); Simulation-Based Calibration.

Resumo

O escalonamento de pessoal é um problema operacional central nas organizações de serviços, onde o número de trabalhadores necessário varia ao longo do dia e da semana. Ajustar o pessoal a esta procura variável tem consequências económicas: a subdotação deixa procura por cobrir e degrada o serviço, enquanto a sobredotação desperdiça mão de obra escassa. A dificuldade está em que a procura é incerta, pelo que um horário fixado à partida pode ter mau desempenho quando a procura se realiza.

Estes horários são habitualmente produzidos por modelos determinísticos de Programação Inteira Mista (*MIP*) que otimizam contra um perfil de necessidade nominal tratado como certo. Quando a procura realizada se afasta desse perfil, sobretudo na sua cauda direita, o horário resultante pode deixar falhas de cobertura significativas, não por falta de trabalhadores ou fraqueza do otimizador, mas porque um único horário fixo, construído sobre uma única necessidade fixa, não absorve a variabilidade em torno dela.

Esta dissertação investiga se é possível tornar um MIP determinístico mais robusto à incerteza da procura sem o redesenhar, atuando apenas sobre a necessidade que lhe é fornecida. Propõe-se um *demand buffer* que inflaciona o perfil nominal com base na variabilidade histórica da procura, em duas especificações: um buffer global com um único nível de conservadorismo, calibrado por pesquisa exaustiva, e um buffer por blocos, mais fino, calibrado por otimização Bayesiana. Os parâmetros são calibrados sobre cenários de treino através de uma camada baseada em simulação e avaliados fora da amostra a dois níveis de carga da força de trabalho.

Com uma força de trabalho fixa, o buffer reduz a falha de cobertura fora da amostra em 11,7% quando a força de trabalho está muito carregada e 49,1% quando tem folga, sem acrescentar capacidade, ao realocar um orçamento de cobertura constante para os períodos de maior variabilidade; o risco de cauda diminui quase tanto como a média, pelo que o ganho não envolve qualquer compromisso entre média e cauda. O buffer por blocos, mais fino, produz ganhos nalgumas janelas mas não melhora de forma fiável o buffer global, em média. A robustificação por *demand buffer* é, assim, uma forma eficaz e de baixo custo de tornar um escalonador determinístico mais robusto à procura estocástica, mais valiosa quando a força de trabalho não está fortemente carregada.

Palavras-chave: Escalonamento de Pessoal; Incerteza na Procura; Robustificação de Horários; Programação Inteira Mista (*MIP*); Calibração por Simulação.

Acknowledgements

Um sincero agradecimento àqueles que me orientaram ao longo de todo este projeto, e que viram potencial nele mesmo quando nem eu conseguia. Isto só foi possível graças à vossa incansável disponibilidade, paciência e orientação: Obrigada.

A todos aqueles que, durante estes cinco anos, proferiam a expressão "Não se agradece"quando eu dizia "Obrigada"ou alguma das suas variantes (sabe-se lá porquê!): Obrigada. Vou ser muito feliz a contar aos meus filhos todas as histórias que passei convosco – todas as noitadas na Sala de Gestão, nas Catacumbas ou noutros sítios cujo nome pode não se adequar a uma dissertação de Mestrado; todos os ataques de riso nos momentos mais inusitados; e todos os momentos que só conseguimos ultrapassar por nos termos apoiado uns aos outros.

Às três pessoas responsáveis pela minha génese, que me viram e fizeram crescer durante estes corridos 23 anos (um deles, apenas 20 deles) e aos quais eu raramente refiro o quão grata sou por tudo aquilo que eles me ensinam, todos os dias: O-bri-ga-da.

Por fim, à pessoa que aceitou apoiar-me incondicionalmente para o resto da minha vida (espero que já saibas o que é um buffer): Obrigada.

Por ter todas estas pessoas ao meu lado, sou então obrigada a dizer: "Obrigada".

"Esperávamos, desejávamos, conseguimos."

Marcelo Rebelo de Sousa

Contents

1 Introduction **1**
- 1.1 Objectives and Approach . . . 2
- 1.2 Contributions . . . 3
- 1.3 Structure of the dissertation . . . 3

2 Literature Review **5**
- 2.1 Deterministic staff scheduling models . . . 5
- 2.2 Uncertainty and the limitations of deterministic schedules . . . 7
- 2.3 Stochastic programming and proactive robustness . . . 8
- 2.4 Robustification mechanisms for deterministic models . . . 10
- 2.5 Simulation-optimization and parameter calibration . . . 12
- 2.6 Research gap . . . 13

3 Problem Description **15**
- 3.1 Operational context and planning process . . . 15
- 3.2 Problem data and scheduling rules . . . 16
- 3.3 Deterministic model . . . 17

4 Proposed Methodology **21**
- 4.1 Simulation–optimisation framework . . . 22
- 4.2 Deterministic optimisation layer . . . 24
- 4.3 Stochastic evaluation layer . . . 26
- 4.4 Total Coverage Constrainedness as an experimental dimension . . . 27
- 4.5 Robustification mechanism . . . 29
  - 4.5.1 Objective-level regularisation as a common preparatory mechanism . . . 29
  - 4.5.2 General buffer formulation and confidence-bound logic . . . 30
  - 4.5.3 Global and block-specific buffer specifications . . . 31
- 4.6 Calibration of demand-buffer configurations . . . 35
  - 4.6.1 Calibration objective . . . 35
  - 4.6.2 Grid search calibration . . . 35
  - 4.6.3 Bayesian optimisation calibration . . . 36
  - 4.6.4 Selection of calibrated configurations . . . 37
- 4.7 Evaluation metrics and benchmark . . . 37

5 Experimental Study Design **41**
- 5.1 Experimental instance construction . . . 41
- 5.2 Demand environments and TCC-controlled scaling . . . 42
- 5.3 Historical demand characterisation . . . 43
- 5.4 Uncertainty approximation for buffer construction . . . 45

5.5 Scenario generation and train–test split 46
5.6 Buffer configuration and calibration settings 47
5.7 Computational settings and experimental matrix 51

6 Experimental Results and Discussion **53**
6.1 Baseline behaviour and stochastic benchmark 53
6.1.1 Deterministic schedule behaviour 53
6.1.2 Stochastic baseline performance 54
6.2 Global demand buffer results 56
6.2.1 Global demand buffer calibration 56
6.2.2 Global demand buffer out-of-sample performance 58
6.3 Block-specific demand buffer 59
6.4 Cross-condition comparison and discussion 59
6.4.1 How the buffer reallocates the fixed workforce 60
6.4.2 Effect of constrainedness 60
6.4.3 Global versus block-specific buffer 64
6.4.4 How much room remains: a non-anticipative benchmark 65
6.4.5 Robustness: solver gaps and test difficulty 66
6.4.6 Average versus tail risk 66

7 Conclusions and Future Work **69**
7.1 Conclusions 69
7.2 Limitations 70
7.3 Future work 71

Bibliography **73**

Sustainable Development Goals **77**

A Relative difficulty of the training and test sets **79**

B Distributional Approximation Diagnostics **83**

C Per-window Results by TCC Level **89**

D Solver Diagnostics and Optimality Gaps **93**
D.1 Incumbent stabilisation and the comparability of time budgets 93

E The non-anticipative ceiling: a sample-average benchmark **95**
E.1 Purpose and hypothesis 95
E.2 Method 95
E.3 Results 96
E.4 What the comparison establishes 96

Acronyms and Symbols

The following lists summarise the acronyms and the main symbols used throughout this dissertation. Symbols are grouped by role; the sub- and superscript conventions are listed at the end.

Acronyms

BO	Bayesian Optimisation
MIP	Mixed-Integer Programming
TCC	Total Coverage Constrainedness

Symbol	Description
Sets and indices	
T	Set of workers, indexed by t
D	Set of days in the planning horizon, indexed by d
P	Set of intra-day periods, indexed by p
W	Set of weekly blocks, indexed by w; $D_w \subseteq D$ are the days of week w
$\mathscr{B}$	Set of temporal blocks (block-specific buffer), indexed by b; $b(d,p)$ is the block of (d,p)
D^{Ψ}	$\{0,\dots,\lvert D\rvert-1-\Psi\}$: window starts for the consecutive-days rule
D^{+}	$D\setminus\{0\}$: days with a preceding day in the horizon
P_{δ}	$\{0,\dots,\lvert P\rvert-1-\delta\}$: early periods bound by the interday rest
S	Scenario set; S_{cal} calibration, S_{test} test scenarios, indexed by s
k	Target constrainedness (TCC) level, $k \in \{0.7, 0.5\}$
$w(d)$	Weekday type of day d
$h(p)$	Hour to which period p belongs
Requirement and demand profiles	
IDEAL	Nominal staffing requirement profile, IDEAL $= (n_{d,p})$
$n_{d,p}$	Nominal staffing requirement in period p of day d
R	Generic staffing requirement profile supplied to the MIP
$R^{k}_{d,p}$	TCC-scaled nominal requirement at level k
$\widehat{R}(\theta)$, $\widehat{R}_{d,p}(\theta)$	Buffered requirement profile for configuration θ
$\widehat{R}^{k,\Delta}_{d,p}$	Buffered requirement at level k and conservatism Δ
$A_{d,p}$	Historical demand observation
$\widetilde{A}^{k}_{d,p}$	TCC-scaled historical demand observation
$A^{s}_{d,p}$, $A^{s,k}_{d,p}$	Realised demand in scenario s (at level k)

Symbol	Description
Model parameters	
$a_{t,d}$	1 if worker t is available on day d, 0 otherwise
L	Effective work periods per working day
β	Meal break duration (periods)
σ	Working-day span, $\sigma = L + \beta$ (periods)
$c_{\min}$	Minimum work before the meal break (periods)
$c_{\max}$	Maximum continuous work before the meal break (periods)
K	Required working days per week
Ψ	Maximum consecutive working days
δ	Interday-rest offset (periods)
τ_p	Duration of period p
Q	Total available workforce capacity over the planning horizon
λ	Regularisation weight in the objective
ϕ_k	TCC scaling factor for level k ($\phi_{0.7} = 11/19$, $\phi_{0.5} = 11/27$)
$s^{\text{buf},k}_{w(d),p}$	Sample standard deviation of scaled training demand (weekday $w(d)$, period p)
Decision and auxiliary variables	
$x_{t,d}$	1 if worker t works on day d
$y_{t,d,p}$	1 if worker t is on duty in period p of day d
$s_{t,d,p}$	1 if worker t's shift starts in period p of day d
$b_{t,d,p}$	1 if worker t's meal break starts in period p of day d
$g^{-}_{d,p}$	Undercoverage in period p of day d
$g^{+}_{d,p}$	Overcoverage in period p of day d
m	Auxiliary variable bounding the maximum period-level overcoverage
Scheduling engine and coverage	
x	A schedule; $x(R)$ the schedule produced by the MIP from profile R
x^0	Deterministic baseline schedule, $x^0 = x(\text{IDEAL})$
x^θ	Buffered schedule, $x^\theta = x(\widehat{R}(\theta))$
$x^{k,0}$	Baseline schedule at constrainedness level k
Ω	Feasible region of the deterministic MIP
$F(x,R)$	Deterministic objective function
$C_{d,p}(x)$	Scheduled coverage: workers scheduled in period p of day d under x
Buffer and robustification	
θ	Buffer configuration
Δ	Conservatism (confidence-level) parameter; global buffer
$\boldsymbol{\Delta} = (\Delta_b)_{b \in \mathscr{B}}$	Vector of block-level conservatism parameters
Δ^*, Δ^*_{grid}	Selected (calibrated) conservatism level
Δ_{grid}	Grid of candidate conservatism values
$B_{d,p}(\theta)$	Buffer term added to the nominal requirement
$D_{d,p}(\Delta)$	Deviation measure (conservative deviation above $n_{d,p}$)
z_Δ	Standard-normal multiplier, $z_\Delta = \Phi^{-1}((1+\Delta)/2)$
$U^{k,\Delta}_{d,p}$	Uncertainty margin added to the scaled requirement
Evaluation metrics	
$M^s_{d,p}(x)$	Missing coverage in scenario s, period p, day d
$M^s(x)$	Total missing coverage in scenario s

Symbol	Description
$\overline{M}_S(x)$	Average missing coverage over scenario set S
$M_S^{\max}(x)$	Worst-case missing coverage over S
$M_S^q(x)$	q-quantile of missing coverage over S
$H^s(x)$	Missing worker-time in scenario s
$\overline{H}_S(x)$	Average missing worker-time over S
$\text{Improvement}_S(x^\theta)$	Percentage improvement of x^θ over the baseline
$\text{SL}_S(x)$	Service level
$\widehat{J}(\theta)$	Calibration objective (mean criterion over S_{cal})
$J(x^\theta, A^s)$	Scenario-level performance criterion
Probability, operators and conventions	
$\mathcal{N}(\cdot,\cdot)$	Normal distribution (given mean and variance)
$\Phi(\cdot)$, $\Phi^{-1}(\cdot)$	Standard-normal cumulative distribution function and its inverse
$\widehat{F}^k_{w(d),h(p)}$	Empirical cumulative distribution function of demand, for weekday $w(d)$ and hour $h(p)$ at level k
$\mathbb{E}[\cdot]$	Expectation
$Q_{q,S}(\cdot)$	q-quantile operator over set S
$\mathbf{1}[\cdot]$	Indicator function
$\text{round}(\cdot)$, $\lceil\cdot\rceil$	Round to nearest integer; ceiling
$\widehat{(\cdot)}$	Buffered quantity (e.g. $\widehat{R}$)
$\overline{(\cdot)}$	Average over a scenario set (e.g. $\overline{M}$)
$(\cdot)^*$	Selected / calibrated value

List of Figures

4.1 Overview of the proposed methodology 22
4.2 Simulation–optimisation calibration loop used to tune demand-buffer configurations on training demand scenarios. 24
4.3 Conceptual illustration of the demand-buffer mechanism for the global and block-specific specifications 32
4.4 Period-specific buffers under a single conservatism level 34

5.1 Average demand profile over the full year 44
5.2 Hour-level demand variability (training set). 44
5.3 Temporal partition used for the block-specific buffer. 49

6.1 Illustrative deterministic behaviour of the baseline schedule 54
6.2 Baseline coverage against realised demand 55
6.3 Calibration objective against the buffer level 57
6.4 Reallocation of the fixed workforce by the global buffer, for a representative window 61
6.5 How far the buffer's requirement can be raised, by constrainedness level. 63

A.1 Quartile-based outlier counts by weekday and time of day, from 05:00 to 11:40. . 80
A.2 Quartile-based outlier counts by weekday and time of day, from 11:50 to 18:30. . 81
A.3 Quartile-based outlier counts by weekday and time of day, from 18:40 to 01:10. . 82

B.1 Distributional diagnostic for Friday 06:00 84
B.2 Distributional diagnostic for Monday 08:00 84
B.3 Distributional diagnostic for Monday 13:00 85
B.4 Distributional diagnostic for Saturday 06:00 85
B.5 Distributional diagnostic for Saturday 23:00 86
B.6 Distributional diagnostic for Sunday 23:00 86
B.7 Distributional diagnostic for Thursday 13:00 87
B.8 Distributional diagnostic for Tuesday 07:00 87

List of Tables

3.1 Case-specific labour and scheduling rules for the retail service post. 17
3.2 Notation of the deterministic MIP model. 18

5.1 Implemented demand environments. 43
5.2 Temporal blocks used in the block-specific buffer partition. 49
5.3 Buffer calibration settings. 50
5.4 Number of calibration runs per method. 50
5.5 Main experimental settings. 52

6.1 Baseline performance 55
6.2 Grid search versus Bayesian optimisation for the global buffer 58
6.3 Out-of-sample performance of the calibrated global buffer on the test scenarios, averaged over the six windows, by constrainedness level. 58
6.4 Out-of-sample performance of the calibrated block-specific buffer on the test scenarios, averaged over the six windows, by constrainedness level. 59

7.1 Contribution of this dissertation to the Sustainable Development Goals 77

A.1 Demand level and frequency of extreme observations in the two sets. 79

C.1 Per-window calibration of the global buffer, by constrainedness level 89
C.2 Per-window out-of-sample performance at TCC 0.7 89
C.3 Per-window out-of-sample performance at TCC 0.5 90
C.4 Per-window calibration of the block-specific buffer 90
C.5 Per-window out-of-sample performance of the block buffer at TCC 0.7 91
C.6 Per-window out-of-sample performance of the block buffer at TCC 0.5 91

D.1 Per-window solver diagnostics at TCC 0.7 93
D.2 Per-window solver diagnostics at TCC 0.5 94
D.3 Incumbent stabilisation across grid and global solves 94

E.1 Sample-average benchmark against the baseline and the two buffers, per window and constrainedness level 96

Chapter 1

Introduction

Staff scheduling is a central operational problem in service organisations, which dominate modern economies: services account for about three quarters of gross value added and employment in Portugal (OECD, 2026; World Bank, 2026). These operations are labour-intensive yet run on thin margins: personnel costs are around 20% of total operating costs across Portuguese firms (Manteu et al., 2021), while the net sales margin in food retail was about 1% in 2023 (Direção-Geral de Economia, 2026). A margin this thin makes staffing errors costly in both directions: understaffing loses sales the margin cannot absorb, and overstaffing adds labour cost that, at a fifth of operating expenses, is itself large. The understaffing side is measurable: in a large retail chain, understaffing during peak hours was estimated to reduce sales by 6.15% and profit by 5.74% (Mani et al., 2015).

The number of workers a service post needs changes throughout the day and across the week, while the workers themselves are bound by contractual and labour rules that limit when and how much they can work. Building a timetable that covers the staffing requirement as closely as those rules and the available workforce allow is, in most settings of practical size, an optimisation problem, and Mixed-Integer Programming (MIP) is one of the established tools for solving it. The resulting formulation, a deterministic staff-scheduling MIP model, is the baseline scheduling engine studied throughout this dissertation.

A deterministic MIP, however, optimises against a single staffing requirement profile that it treats as certain. In the operation studied in this dissertation, a service post of a food retailer, that profile is the organisation's own nominal staffing plan produced by an internal forecasting process. It is supplied to a deterministic MIP scheduler that returns a schedule minimising the shortfall against it, that is, the gap between the staffing requirement and the workers actually scheduled. Realised demand, though, is not certain: it diverges from the nominal plan period by period, and in particular its right tail rises above the planned level. A schedule that is optimal for the nominal requirement can therefore leave substantial shortfall once demand is actually realised, not because the workforce is too small or the optimiser too weak, but because a single fixed schedule built on a single fixed requirement cannot absorb the variability around it.

The natural response would be to replace the deterministic model with a stochastic one. This dissertation pursues a lighter alternative, for two reasons. The first is specific to this setting:

the nominal requirement is produced by a forecasting process that does not see the labour and operational constraints of the scheduler, while the scheduler optimises those constraints without representing demand variability; reshaping the requirement connects the two, letting demand-variability information enter a constraint-aware optimiser without rebuilding it. The second is computational: at the scale and constraint density of real rostering instances, a fully stochastic reformulation is costly to build and hard to integrate, and these costs grow as the formulation is made more realistic. Rather than redesigning the model, the dissertation acts on the input the model receives, reshaping the nominal requirement into a buffered requirement that anticipates demand variability, so that the schedule the unchanged model returns is more robust to the demand it will actually face, while retaining the planning forecast rather than discarding it. This reshaping is what the dissertation calls a demand buffer.

1.1 Objectives and Approach

The objective of this dissertation is to determine whether demand-buffer robustification can improve the out-of-sample stochastic performance of a deterministic staff-scheduling model without changing the model, and under what conditions. This general aim is addressed through five research questions.

1. Does reshaping the nominal requirement with a demand buffer reduce out-of-sample staffing shortfall, in expectation and in the tail, relative to the deterministic baseline?
2. How does any benefit depend on how tightly the workforce is loaded relative to demand?
3. Through what mechanism does the buffer act, given that workforce capacity is fixed?
4. Does a finer, block-specific buffer outperform a single global buffer?
5. Does the calibrated buffer leave room for improvement?

The deterministic MIP scheduler is treated throughout as a fixed engine, held unchanged by design, and the study acts only on the requirement profile supplied to it. The demand buffer raises the nominal requirement using demand variability estimated from historical data, in two specifications of increasing temporal granularity: a global buffer with a single conservatism level for the whole horizon, calibrated by grid search, and a block-specific buffer with separate levels for a small number of temporal blocks, calibrated by Bayesian optimisation. Buffer parameters are selected on training scenarios through a simulation-based calibration layer, and the resulting schedules are evaluated out of sample on held-out test scenarios, so that any difference in performance reflects the schedule rather than the conditions under which it is assessed. The analysis is carried out at two levels of workforce constrainedness, so that the effect of how tightly the workforce is loaded can be examined directly, and the calibrated buffers are compared against a non-anticipative benchmark, fitted to the historical demand distribution, that bounds how much any fixed schedule can achieve.

1.2 Contributions

The dissertation makes three main contributions.

First, it formulates demand-buffer robustification: a general buffered-requirement formulation, with global and block-specific specifications, that makes a deterministic staff-scheduling model robust to demand uncertainty without reformulating the optimisation model or adding capacity.

Second, it develops a simulation-based calibration framework that tunes the buffer against out-of-sample stochastic performance, using grid search for the global specification and Bayesian optimisation for the block-specific one.

Third, it reports an extensive computational study of the resulting schedules under stochastic demand at two levels of workforce constrainedness. The study shows that, on a fixed workforce, the buffer reduces both expected and tail shortfall by reallocating a constant coverage budget towards the more variable periods, and that the size of this benefit depends on how tightly the workforce is loaded, with the mechanism behind that dependence identified. It also evaluates whether finer temporal granularity helps, finding no dependable advantage of a block-specific buffer over the global one on these data.

1.3 Structure of the dissertation

The remainder of this dissertation is organised as follows. Chapter 2 reviews staff scheduling as an optimisation problem and the treatment of demand uncertainty within it. Chapter 3 describes the operational problem and the deterministic MIP used as the baseline scheduling engine. Chapter 4 develops the demand-buffer methodology, including the buffer formulation, its global and block-specific specifications, and the simulation-based calibration and evaluation layers. Chapter 5 sets out the experimental design: the planning windows, the demand scenarios, the constrainedness levels, the temporal partition and the computational settings. Chapter 6 presents and discusses the results, from the deterministic baseline through the global and block-specific buffers to the cross-condition analysis and the non-anticipative benchmark. Chapter 7 draws the conclusions, states the limitations of the study, and outlines the work they motivate.

Chapter 2

Literature Review

This chapter reviews the literature relevant to robustifying deterministic staff scheduling models under uncertainty. It first introduces staff scheduling as an optimisation problem, focusing on deterministic mixed-integer programming formulations, coverage requirements, and labour constraints. It then examines the limitations of deterministic schedules when demand, workforce availability, and arrival patterns deviate from nominal assumptions. The chapter then reviews stochastic and robust approaches to address uncertainty, before turning to proactive robustness mechanisms, especially capacity buffers and demand protection. Finally, it discusses the role of simulation-optimisation and parameter calibration in evaluating and tuning such mechanisms under stochastic conditions. The chapter concludes by synthesising these threads and identifying the research gap this dissertation addresses.

2.1 Deterministic staff scheduling models

In the academic literature, personnel scheduling (often called rostering) is the process of building work timetables that let an organisation meet its service or production requirements within the contractual and legal constraints that apply to its workers (Ernst et al., 2004). It is usually carried out in stages rather than as a single decision: demand modelling first converts expected workload into per-period staffing requirements, and later stages assign individual workers to specific shifts or tasks (Ernst et al., 2004; Van den Bergh et al., 2013). Cost is a large part of why it matters, since labour is typically one of the biggest components of an organisation's direct operating costs (Van den Bergh et al., 2013). This is especially true in service industries, where staffing decisions affect both daily operational productivity and the quality of service delivered to customers (Ağralı et al., 2017).

A defining feature of the problem is its combinatorial nature. Schedules are built from discrete assignments of employees to shifts, subject to numerous interacting requirements, so the number of candidate solutions grows rapidly with problem size. Staff scheduling is generally classified as NP-hard (Ağralı et al., 2017; Maheut et al., 2024), meaning the computation needed to guarantee an optimal solution grows steeply as more employees, shifts, and constraints are added. In settings

such as healthcare or retail, this scale makes manual scheduling impractical and has driven the adoption of automated optimisation methods (Ernst et al., 2004; Maheut et al., 2024).

Among these methods, Mixed-Integer Programming (MIP) is widely used for deterministic staff scheduling. Its appeal comes from the natural match between binary variables and discrete assignment decisions (e.g. whether or not an employee is assigned to a given shift), and from its ability to handle, within a single model, the rigid constraints typical of deterministic settings (Warner and Prawda, 1972; Billionnet, 1999; Van den Bergh et al., 2013). A further advantage is that such formulations can be implemented in established modelling languages and solved with general-purpose solvers. This makes obtaining optimal or near-optimal solutions tractable in practice (Billionnet, 1999).

Within these formulations, coverage requirements are a central modelling element. Van den Bergh et al. (2013) report that the coverage constraint is among the most frequently encountered features in the literature, appearing as a hard constraint in roughly 75% of the studies they review. The same authors note, however, that many models also express coverage through soft constraints by category, to allow some flexibility in how requirements are met. Models often achieve this by relocating coverage to the objective function. Maheut et al. (2024), for instance, combine a hard minimum staffing requirement with minimisation of excess personnel in the objective, which discourages idle time and unnecessary labour cost; Ağralı et al. (2017) move the coverage requirement into the objective to satisfy the maximum weighted number of service requirements rather than impose absolute coverage on every task.

Labour constraints complement the coverage requirements by bounding the space of feasible schedules. These typically include limits on working hours, minimum rest periods between shifts, restrictions on consecutive working days, and individual availability (Ağralı et al., 2017; Maheut et al., 2024). Such conditions are modelled as hard constraints: violating them would render a schedule contractually or legally invalid, so they define feasibility and are not traded off against cost.

The choice of objective function shapes how coverage is treated. Deterministic models generally minimise labour cost while penalising deviations from desired staffing levels; the result is a trade-off between understaffing and overstaffing. Understaffing is typically modelled as a shortage cost for unmet service requirements (Warner and Prawda, 1972), while overstaffing incurs avoidable labour costs and idle capacity (Maheut et al., 2024). Several formulations address both simultaneously: Parisio and Jones (2015) minimise overstaffing and understaffing jointly, and Ağralı et al. (2017) minimise unmet demand directly.

All of these formulations share the assumption that staffing requirements are known in advance. Deterministic models treat demand as a fixed, nominal input, typically derived from forecasts or historical averages (Parisio and Jones, 2015), and optimise a single schedule against it. This is what makes the problem tractable, but it is also the main limitation: the schedules produced are only as reliable as the demand estimates on which they rest. When actual conditions differ from those estimates, an otherwise optimal schedule may perform poorly in practice (Parisio and Jones, 2015). Deterministic approaches tend to lose reliability when operating conditions vary in

the short term (Ingels and Maenhout, 2019).

2.2 Uncertainty and the limitations of deterministic schedules

Once staffing requirements are treated as uncertain rather than fixed, the limitations of deterministic schedules become more apparent. The literature typically groups this variability into three sources: demand uncertainty, capacity or availability uncertainty, and arrival-time uncertainty (Van den Bergh et al., 2013). Together, these sources cover most of the gap between planned and actual operating conditions.

Demand uncertainty arises because service demand is volatile and hard to forecast accurately, meaning point estimates often miss the underlying variability (Parisio and Jones, 2015). It appears as fluctuations in customer volume and deviations in peak periods, the latter sometimes called peak shifting (Parisio and Jones, 2015; Mattia et al., 2017). Forecasting errors translate into staffing levels that prove inadequate once demand materialises (Parisio and Jones, 2015).

Capacity, or availability, uncertainty is about whether the workforce assumed at the planning stage is actually available during operations. Employee absence and short-term variation in availability mean that actual capacity may fall short of the rostered level, widening the gap between planned and realised staffing (Ingels and Maenhout, 2019; Van den Bergh et al., 2013).

Arrival-time uncertainty, a third source, is about when work arrives rather than how much: where operations face stochastic delays, work may arrive later or less evenly than anticipated, leaving planned capacity for a given period momentarily insufficient even when the daily total is correct (De Bruecker et al., 2015).

These overlapping sources create a mismatch between planned capacity and realised workload. Because deterministic models optimise a single schedule against nominal or historically averaged requirements, optimising against the mean yields a biased, often optimistic estimate of a schedule's true cost (Robbins and Harrison, 2010). The mismatch extends beyond daily totals: even when the total hours for a day are correct, poor intraday distribution from forecast error can produce simultaneous periods of shortage and surplus (Ingels and Maenhout, 2019; Mattia et al., 2017). Divergences between service demand and staff availability thus cause both shortages and overstaffing (Ingels and Maenhout, 2019). This dissertation addresses the first of these uncertainty sources, demand uncertainty.

Operationally, this mismatch manifests as understaffing or overstaffing, and the two are generally treated as qualitatively different problems. Overstaffing principally raises labour costs and may reduce workforce satisfaction through under-occupation (Mattia et al., 2017). Understaffing, by contrast, reduces service levels and bears directly on revenue (Mattia et al., 2017); it also carries a less visible reputational cost from perceived poor service quality (Robbins and Harrison, 2010).

The two are not, however, symmetric in their consequences. In many settings, the cost of failing a service-level commitment or losing a sale exceeds the marginal cost of an additional worker (Robbins and Harrison, 2010; Mattia et al., 2017). This imbalance is most pronounced where the contribution margin per unit of service is high: the opportunity cost of an unserved customer can

substantially exceed the cost of employing one more worker, which leads robust formulations to recommend higher staffing levels than a purely cost-minimising deterministic model would (Easton, 2014). A further systemic dimension arises from endogenous absenteeism: because employees may be averse to high anticipated workloads, understaffing can itself raise absence rates, setting off a self-reinforcing cycle of service degradation that overstaffing does not (Green et al., 2013). These arguments suggest that understaffing is generally the more consequential deviation and justify planning for some surplus capacity, which raises wage costs but helps absorb unexpected events; schedules optimised around average demand, by contrast, tend to understate the recovery costs incurred when understaffing actually materialises (Ingels and Maenhout, 2019; Robbins and Harrison, 2010).

Demand variability is thus a primary driver of the gap between expected and realised schedule performance. However, uncertainty has frequently been simplified or omitted from scheduling models to preserve tractability, and explicit treatment of it remains comparatively rare (Ingels and Maenhout, 2019; Van den Bergh et al., 2013). This gap motivates stochastic, robust and simulation-based approaches.

2.3 Stochastic programming and proactive robustness

Stochastic programming replaces the single nominal demand of a deterministic model with multiple scenarios and optimises expected performance across them rather than against a single average (Parisio and Jones, 2015).

A common approach is the two-stage formulation with recourse. First-stage decisions, such as the assignment of employees to weekly schedules, are fixed before demand is known; second-stage recourse actions then adjust the plan once uncertainty is realised, to correct under- or over-allocation (Parisio and Jones, 2015). This structure explicitly acknowledges that some adjustment will be necessary in practice, and assumes that such adjustment will be possible; recourse actions have been shown to lower long-run average costs compared with plans that ignore adjustment (Parisio and Jones, 2015).

A related strand expresses schedule reliability directly through a service-level requirement that the roster must satisfy under uncertainty. De Bruecker et al. (2015) embed such a stochastic service-level constraint within a model-enhancement heuristic that iteratively adjusts a mixed-integer linear programme towards a stochastic environment using simulation output. Their results indicate that this can secure a desired service level at an acceptable increase in labour cost (De Bruecker et al., 2015). The approach is a pragmatic middle path: it retains a deterministic optimisation core while using simulation to guide it, rather than formulating a fully stochastic programme.

Across these formulations, under- and overstaffing are addressed by penalising deviations across scenarios, permitting recourse, or imposing service-level targets, rather than by assuming a single schedule will match realised demand (Abernathy et al., 1973; Parisio and Jones, 2015). Where such methods have been evaluated, they reduce the combined cost of under- and over-allocation relative to expected-value or deterministic solutions; reported improvements reach around

6% in retail scheduling and 10% in maintenance scheduling in their respective performance metrics (Parisio and Jones, 2015; Duffuaa and Al-Sultan, 1999). These figures are specific to their respective contexts and should not be read as a general guarantee that stochastic models outperform deterministic ones.

These advantages come at a cost. The scheduling problem is already computationally demanding: even in its deterministic form it is NP-hard (Ağralı et al., 2017; Maheut et al., 2024). Introducing scenarios compounds this difficulty: representing uncertainty adequately may require many scenarios, so that scenario generation and reduction, often handled through clustering to retain only the most significant cases, become a practical challenge in their own right (Parisio and Jones, 2015). In richer settings, integrating uncertainty into models already burdened by labour regulations and business rules can render exact solution intractable, which is one reason the stochastic component is sometimes simplified or omitted in practice (Parisio and Jones, 2015).

A final limitation is conceptual rather than computational. Protecting a schedule against uncertainty generally entails additional labour cost, so the central challenge becomes calibration: balancing the desired service level against the cost of the protection needed to achieve it (De Bruecker et al., 2015). If this balance is not managed, robustness can tip into excessive conservatism (De Bruecker et al., 2015). Partly because fully stochastic models are difficult to solve at the scale of real rostering instances, a complementary line of work pursues robustness more pragmatically, building protective slack into the roster itself rather than reformulating the optimisation around the full distribution of outcomes (De Bruecker et al., 2015; Restrepo et al., 2017). This motivates a shift towards proactive robustness mechanisms, where protection is built into the roster before uncertainty is realised.

In the rostering literature, a robust roster combines stability and flexibility when disruptions occur, with sufficient absorption capacity to limit the reactive changes needed (Ingels and Maenhout, 2019). Robustness is thus framed as the ability to withstand variability while minimising reactive intervention, rather than as optimality under a single forecast. A related, performance-oriented view defines stochastic robustness as a roster's ability to guarantee a target service level despite delays or uncertainty (De Bruecker et al., 2015). The two facets are complementary: stability concerns how little a published roster must be altered (Ingels and Maenhout, 2019, 2018), while flexibility concerns how well it can adapt when alteration is unavoidable (Ingels and Maenhout, 2018).

Stability matters because changes made after a roster has been published are operationally and socially costly: they disrupt employees' personal lives and reduce satisfaction (Ingels and Maenhout, 2019; Doneda et al., 2026). Moreover, a roster optimised purely for planned cost tends to leave little room to absorb disruptions, so that costs realised in operation can exceed those anticipated at the planning stage (Ingels and Maenhout, 2015). Stability is therefore not equivalent to satisfying deterministic coverage requirements: a roster may meet nominal coverage exactly yet remain fragile, since feasibility under expected conditions reveals little about behaviour once those conditions shift (Ingels and Maenhout, 2018, 2019, 2015).

2.4 Robustification mechanisms for deterministic models

Having established proactive robustness as an alternative to full stochastic reformulation, the discussion now turns to the mechanisms through which such robustness can be embedded in deterministic rosters. A useful conceptual lens for this topic is the parametric Cost Function Approximation framework, in which a parameterised version of a deterministic optimisation model is treated as a practical way of handling uncertainty rather than as an ad hoc industrial heuristic (Powell and Ghadimi, 2022). Within this framework, quantities such as schedule slack or buffer stock are structured parameters added to a deterministic model so that it can cope with variability without building the scenario trees characteristic of full stochastic programming (Powell and Ghadimi, 2022). Powell distinguishes two general ways of introducing such parameters: through the constraints of the model, for example as slack or reserve capacity that protects feasibility, or through its objective, as bonuses or penalties that discourage poor service (Powell, 2022).

Within personnel rostering specifically, capacity buffers are the most direct expression of this idea. By introducing spare capacity into the shift roster before uncertainty is realised, they improve stability and reduce the reactive changes subsequently needed (Ingels and Maenhout, 2019). Because this slack is proactive rather than reactive (Ingels and Maenhout, 2018, 2015), larger-than-expected demand, absences or other operational variation can be absorbed within the existing plan rather than through last-minute correction (Doneda et al., 2026). The literature uses several terms for this protective capacity (slack, reserve, buffer stock); this dissertation refers to it collectively as a buffer.

This proactive buffer takes several operational forms: reserve duties planned into the roster and converted to active duty when disruptions arise (Ingels and Maenhout, 2015; Doneda et al., 2026), prescheduled proactive overtime (Ingels and Maenhout, 2018), and greater employee substitutability through overlapping skills (Ingels and Maenhout, 2017).

The value of a buffer depends on how much is provided and where it is placed, and the literature treats these as distinct questions: deciding where to position protective capacity draws on domain knowledge and problem structure, whereas the size is a parameter to optimise rather than fix arbitrarily (Powell and Ghadimi, 2022). Allocating buffers by intuition alone is difficult, and arbitrary slack risks being insufficient where needed and wasteful where it is not (Ingels and Maenhout, 2019). A more recent strand therefore determines buffer size and position endogenously, treating them as decision variables that match the added capacity to the structure of stochastic demand (Ingels and Maenhout, 2019).

This positioning problem reflects an underlying trade-off between robustness and cost. A larger buffer reduces staff shortages but raises wage costs and the cost of capacity that ultimately goes unused (Ingels and Maenhout, 2019); robust formulations are correspondingly described as securing service levels at an acceptable increase in labour cost (De Bruecker et al., 2015). In Powell's terms, this amounts to tuning a parameter that trades resource productivity, which favours smaller buffers, against the penalty for service failures, which favours larger ones (Powell, 2022). Robustness must therefore be balanced against efficiency, and assessed rather than assumed. In

practice, the method simulates demand and capacity uncertainty to compare planned with realised performance and measure the adjustment needed to restore feasibility; proactive rosters are then benchmarked against deterministic, minimum-cost baselines that ignore uncertainty (Ingels and Maenhout, 2019, 2015). Bounding mechanisms such as deviation measures or an uncertainty budget can also cap the permissible deviation from expected staffing requirements (De Bruecker et al., 2015). The effectiveness of any buffer, however, ultimately depends on how demand uncertainty translates into concrete buffer requirements.

The basic approach is to redefine or inflate those requirements. Instead of optimising against the expected, or nominal, staffing level, the model targets a higher, protected level that already contains an allowance for variability (Ingels and Maenhout, 2017). The distinction matters. The nominal requirement reflects what is expected on average, whereas the buffered requirement reflects the coverage target once variation is accounted for. These protected requirements can be specified to include reserve duties or additional skilled staff above the minimum, with the extra capacity accounting for variability expected during operational allocation (Ingels and Maenhout, 2017).

How the allowance is computed varies considerably. The simplest approach applies a fixed ratio or percentage uniformly, but the literature also describes more targeted methods based on deviation measures, uncertainty budgets, and upper-bound or service-level requirements (Ingels and Maenhout, 2019). A statistical route derives the margin from the demand distribution itself: Jongbloed and Koole (2001) treat the arrival rate as a random variable and apply the Erlang formula to the upper bound of a prediction interval, tying the margin to a chosen protection level rather than an arbitrary multiplier. Distributionally robust formulations follow a comparable logic without committing to a single distribution: a dispersion model, such as a chi-squared statistic, defines a confidence region for the true probabilities, and safety factors are added to the nominal requirement to guard against additional understaffing (Liao et al., 2013).

Whatever the method, the size of the buffer has to be calibrated. Too small a buffer fails to protect the roster; too large a buffer raises cost and commits capacity that is not needed (Ingels and Maenhout, 2019). The robust optimisation literature puts this more sharply. If the uncertainty set is too large, the solution becomes very conservative and its objective value is far worse than the nominal one, and in practice adequate protection is often achieved with immunisation factors well below what theory would suggest (Liao et al., 2013). Buffers should therefore be tuned or bounded rather than fixed by habit.

Placement matters as much as size. Spreading protection evenly across every period is wasteful, because risk is not evenly distributed, and buffers are more effective when concentrated in the shifts and days where uncertainty is greatest (Ingels and Maenhout, 2019). Uncertainty budgets and immunisation factors serve a related purpose, capping how much protection is admitted in total and so preventing the schedule from being protected everywhere at once (Liao et al., 2013).

A buffer, however, does not create capacity. Headcount, contracts, and skill mix are settled in the staffing phase, a longer-term budgeting decision that constrains the scheduling options available (Maenhout and Vanhoucke, 2013). When the workforce is fixed, a buffer can only reserve

or redistribute existing capacity, for example by holding part of it as reserve duties (Ingels and Maenhout, 2015). Whether there is room to do so depends on the instance. The Total Coverage Constrainedness (TCC) indicator measures the ratio between aggregated staffing requirements and the theoretical maximum number of possible assignments (Ingels and Maenhout, 2017). A high TCC means most available capacity is already committed to nominal requirements, which leaves little freedom to position buffers. The specific form used in this dissertation is defined in Equation (4.6). When constrainedness is high and the desired buffer is large, time-related constraints leave too little slack for buffers to be placed effectively, and stability suffers (Ingels and Maenhout, 2017). The value of a buffer is therefore conditional on how much structural slack the instance contains.

2.5 Simulation-optimization and parameter calibration

The conditional value of buffers creates an evaluation problem. A roster that appears adequate against nominal staffing requirements may perform differently once demand, attendance or workload vary in operation. For this reason, the literature often uses simulation as an evaluation layer: optimisation produces a candidate schedule, while simulation exposes that schedule to stochastic operating conditions and estimates its realised performance (Avramidis et al., 2010; Miranda et al., 2018; Chen et al., 2023). In this role, simulation does not replace the optimisation model; rather, it tests whether the solution generated by the model remains effective when the assumptions embedded in the nominal formulation are relaxed.

Chen et al. (2023) make this explicit for service systems, where what matters is the economic value of a solution, captured for example by an expected opportunity cost rather than whether the nominal selection was correct. The same study, set in a hospital emergency department, uses simulation to evaluate allocations that improve a secondary service objective while still satisfying mandatory probabilistic requirements (Chen et al., 2023). This evaluation exposes the gap between planned and realised performance: a schedule that looks adequate against expected requirements may perform differently once uncertainty materialises (Frantzén et al., 2011).

This pairing of the two methods has a name. Hybrid simulation-optimisation combines an optimisation model with a simulation model, and the literature classifies these hybrids by the role the simulation plays. Figueira and Almada-Lobo (2014) distinguish solution evaluation, where the simulation judges solutions the optimiser produces, from analytical model enhancement, where simulation improves the optimisation model itself. The two roles are not exclusive, but they separate simulation as a judge from simulation as a calibrator (Figueira and Almada-Lobo, 2014). The methodology developed in this dissertation belongs to the second role: simulation is used to calibrate a demand buffer that enhances the deterministic scheduling model, not to select the deployed schedule among candidates.

Simulation can reveal where a deterministic model performs poorly under uncertainty and feed that information back into the model. One documented case is a fare-collection shift-scheduling application in which simulation estimates the expected service level of a candidate roster; when

it falls short of a predefined standard, the method adds a constraint to the integer programme and iterates until the standard is met (Miranda et al., 2018). A comparable feedback loop appears in multiskill call-centre scheduling, where simulation and mathematical programming work together to revise staffing decisions based on simulated service levels (Avramidis et al., 2010). In both cases, the deterministic model is corrected against stochastic feedback rather than discarded.

That correction can be expressed through the model's parameters, which raises the question of how their values should be set. Powell and Ghadimi (2022) argue that the parameters of a parameterised model cannot be chosen reliably from the nominal deterministic model alone and must be tuned against performance under uncertainty, for instance in a stochastic simulator. The cost of getting this wrong runs in both directions: a protection parameter set too high makes the policy overcautious, while one set too low leaves work uncovered (Powell and Ghadimi, 2022). Buffers, reserves and penalties are therefore useful only to the extent that they are calibrated, since their value cannot be judged from the deterministic objective alone (Powell, 2022).

Calibration of this kind is expensive. Combining simulation and optimisation is computationally demanding, so their interaction must be designed carefully (Figueira and Almada-Lobo, 2014). Each candidate parameter setting may require solving an optimisation model and running enough simulation replications to estimate its performance, and exhaustively evaluating every setting is impractical when each evaluation is slow and noisy (Chen et al., 2023). This motivates sample-efficient search, using algorithms that can optimise with few evaluations (Amaran et al., 2016). Bayesian optimisation is often presented as suitable for expensive black-box evaluations of this type because it uses prior results and the correlations between scenarios to decide where to sample next (Chen et al., 2023). In its common form, it builds a Gaussian process (or Kriging) surrogate of the response and selects the next setting using an expected-improvement criterion that balances exploration against exploitation (Amaran et al., 2016). This approach has also been used to tune the parameters of complex control systems automatically (Ramanujam and Li, 2025). Within this taxonomy, the search algorithm is orthogonal to the modelling choice: Bayesian optimisation can serve either role, so the decision to enhance the analytical model with a calibrated buffer shapes the method more than the search procedure used to calibrate it (Figueira and Almada-Lobo, 2014).

Taken together, the literature describes a workflow rather than a single technique. Optimisation generates candidate schedules, simulation evaluates their performance under uncertainty, and a calibration procedure searches the parameter space for settings that balance robustness against efficiency. This is the structure on which the demand-buffer methodology of Chapter 4 is built.

2.6 Research gap

Deterministic mixed-integer programming is a widely used approach to personnel scheduling and rostering, valued for handling discrete assignments and coverage requirements, which appear as the central constraint in most of the literature (Van den Bergh et al., 2013). Such models build a

schedule from nominal staffing requirements derived from forecasts or historical averages. This makes the problem tractable, but it also fixes the schedule to a single expected scenario.

The reviewed literature shows that this assumption is the main weakness of the deterministic approach. Schedules optimised against point forecasts tend to be optimistic and biased, and they degrade once demand or attendance departs from the nominal case (Parisio and Jones, 2015; Robbins and Harrison, 2010). This is what motivates robustification rather than re-solving the nominal model more precisely.

Stochastic and robust formulations respond to this directly and have been reported to improve on expected-value solutions in particular settings (Parisio and Jones, 2015). Their main limitation is practical: at the scale and constraint density of real rostering instances, fully stochastic models become difficult to solve (Restrepo et al., 2017). This has encouraged more pragmatic strategies that adapt a deterministic model rather than replace it (De Bruecker et al., 2015).

Within that pragmatic line, proactive robustness is commonly introduced through capacity buffers or protected staffing requirements, sometimes derived from confidence-interval or chance-constraint reasoning about demand (Ingels and Maenhout, 2019; Jongbloed and Koole, 2001). The literature is consistent that the effect of such buffers depends on how much structural slack the instance contains, and TCC is used to describe that tightness (Ingels and Maenhout, 2017). It is comparatively less explicit, however, on using TCC as more than a descriptor of test instances.

Simulation-optimization provides a natural evaluation layer, since realised service levels often cannot be read from the nominal objective and are estimated through stochastic simulation, with parameters such as buffers treated as tunable quantities (Powell, 2022; Chen et al., 2023). However, this literature is less explicit on how the effectiveness of calibrated parameters depends on the structural characteristics of the underlying scheduling instance (Ramanujam and Li, 2025).

Taken together, these streams are rarely brought together around one question. There remains a need to examine how demand-buffering strategies should be calibrated and evaluated for an existing deterministic MIP under stochastic demand, and how their effectiveness varies with structural constrainedness rather than being applied as fixed or arbitrary margins. This dissertation addresses that gap by robustifying a deterministic MIP through protected staffing requirements, evaluating the resulting schedules under stochastic scenarios, and using TCC as an explanatory lens for when such buffers can be absorbed.

Chapter 3

Problem Description

This chapter describes the operational staff scheduling problem studied in this dissertation and presents the deterministic Mixed-Integer Programming (MIP) model used as the baseline scheduling engine.

Given a staffing requirement profile and workforce availability information, the MIP model returns a feasible schedule according to its objective function and constraints. Its formulation, variables and labour rules are therefore taken as the deterministic baseline for the remainder of the dissertation. In Chapter 4, this baseline is used as the scheduling engine within a robustification framework that modifies the staffing requirement profile through a demand buffer while leaving the underlying MIP unchanged.

The chapter is organised as follows. Section 3.1 introduces the operational context and planning process; Section 3.2 describes the available data and case-specific scheduling rules; and Section 3.3 presents the deterministic MIP model and discusses the operational limitations that motivate the robustification methodology of Chapter 4.

3.1 Operational context and planning process

In the operational setting considered in this dissertation, staff scheduling is the problem of matching available labour capacity with requirements that vary over time. The number of workers a service post needs changes through the day and across the week, while the workers themselves are subject to contractual and labour rules that restrict when and how much they can work. The planning task is therefore to build a timetable that respects these rules while covering as much of the staffing requirement as the available workforce and labour rules allow.

The case examined concerns a large retail store, and focuses on a single service area within it, referred to throughout as the post. The planning process at this post starts from the staffing requirements defined for that post: before any schedule is built, the organisation estimates how many workers the post is expected to need in each period of its operating day. These expected staffing requirements are not uniform. They rise and fall over the course of the day and differ from one day to another, reflecting the organisation's internal assessment of expected activity at the

post. This requirement-setting step is carried out before the schedule is constructed and is taken as given in this dissertation.

The organisation records these expected staffing requirements as a nominal staffing requirement profile, referred to throughout this dissertation as IDEAL, the organisation's own label for it; in the formulation of Section 3.3 the same profile is denoted $n_{d,p}$. For each period of each day in the planning horizon, IDEAL states the ideal number of workers the post should have on duty. At this stage it is an organisational planning requirement: it expresses, period by period, the staffing level the operation aims to reach. The organisation's original planning objective is deterministic: to construct a schedule that leaves as little of this nominal profile uncovered as possible, given the available workforce and the applicable labour rules. When the workers scheduled in a given day and period fall below the IDEAL level, the result is a staffing shortfall; expressed in this dissertation as missing hours, it is measured relative to IDEAL, and reducing it is the criterion that guides the scheduling process.

Against this requirement profile, the scheduling problem is to determine which days each worker works, as well as their shift start and break periods, while respecting the labour rules governing working time and rest. For the purposes of this decision, workers are treated as homogeneous and interchangeable: they are not differentiated by skill, role, seniority or productivity. A period is therefore considered covered in the same way regardless of which available worker is assigned to it.

In this setting, the scheduling problem is solved operationally by a deterministic MIP model. For a given planning horizon (for example, a week or a month), the model takes the requirement profile for that horizon together with workforce availability information and returns a feasible schedule that satisfies the labour and operational rules while minimising the staffing shortfall relative to IDEAL. The same model structure can be applied to different planning horizons and available workforces; these changes define different instances of the baseline model rather than changes to the formulation itself. The schedule it produces is therefore the result of optimising against a single, fixed requirement profile that the model treats as certain.

The next section describes the data and the planning environment from which the requirement profile and workforce information are drawn.

3.2 Problem data and scheduling rules

Having introduced the operational problem and planning process, this section presents the case-specific data and scheduling rules used to instantiate the deterministic problem. These values define the operational setting studied in this dissertation, while the following section abstracts them into a more general mathematical formulation.

The available data correspond to the year 2023 and describe a single retail service post within one operational unit. The post operates on a long daily cycle, from 05:00 until 01:20 of the following day. This operating window is discretised into ten-minute periods, which yields 122

periods per operating day. Every day in the data uses this same sequence of 122 periods, and period-level staffing requirements are represented at this resolution.

The staffing requirement data consist of one IDEAL value for each day-period pair. Each value is an integer headcount: the number of workers the post should have on duty in that period of that day. These values come from the organisation's own forecasting and planning process, which lies outside the scope of this dissertation. The requirement profile is therefore treated as an exogenous input to the scheduling task.

Workforce information is available across the year and identifies which workers can be considered for assignment over time. As established in Section 3.1, the workers are treated as homogeneous and interchangeable, so they differ only in their availability. The main contractual and operational rules that define feasible assignments in this case are summarised in Table 3.1. Each working day includes 44 periods of effective work and a one-hour meal break, which must be placed after a minimum amount of work of two hours has been completed and before the maximum continuous working time of five hours is reached.

Table 3.1: Case-specific labour and scheduling rules for the retail service post.

Scheduling rule	Value	Periods
Effective daily working time	7 h 20 min	44
Meal break duration	1 h	6
Minimum work before meal break	2 h	12
Maximum continuous work before meal break	5 h	30
Minimum interday rest	11 h	66
Working days per week	6 days	
Maximum consecutive working days	6 days	

Although the data cover a full year, the deterministic MIP does not need to be solved over the entire year at once. Any planning horizon selected within 2023, together with the requirement profile and the workforce availability for that horizon, defines a specific planning instance. The next section abstracts these case-specific values into the generic notation of the deterministic MIP model, so that the formulation can be stated independently of the particular horizon and case-specific rule values used.

3.3 Deterministic model

This section abstracts the case-specific values introduced in Section 3.2 into a generic deterministic MIP formulation. Given a nominal staffing requirement profile and workforce availability information over a planning horizon, the model generates a feasible schedule that minimises staffing shortfalls. The formulation is expressed in terms of arbitrary worker sets, planning horizons, and requirement profiles, making it applicable beyond the specific case study considered in this dissertation. The notation used throughout the formulation is summarised in Table 3.2. Using this notation, the generic deterministic MIP model can be formulated as follows.

Table 3.2: Notation of the deterministic MIP model.

Symbol	Description
Sets and indices	
T	Set of workers, indexed by t
D	Set of days in the planning horizon, indexed by d
P	Set of intra-day periods, indexed by p
W	Weekly blocks partitioning D; $D_w \subseteq D$ are the days of week w
D^{Ψ}	$\{0,\ldots,\lvert D\rvert - 1 - \Psi\}$: window starts for the consecutive-days rule
D^{+}	$D \setminus \{0\}$: days with a preceding day in the horizon
P_{δ}	$\{0,\ldots,\lvert P\rvert - 1 - \delta\}$: early periods bound by the interday rest
Parameters	
$n_{d,p}$	Nominal staffing requirement (IDEAL) in period p of day d
$a_{t,d}$	1 if worker t is available on day d, 0 otherwise
L	Effective work periods per working day
β	Meal break duration (periods)
σ	Working-day span, $\sigma = L + \beta$ (periods)
$c_{\min}$	Minimum work before the meal break (periods)
$c_{\max}$	Maximum continuous work before the meal break (periods)
K	Required working days per week
Ψ	Maximum consecutive working days
δ	Interday-rest offset (periods)[a]
Decision variables	
$x_{t,d}$	1 if worker t works on day d
$y_{t,d,p}$	1 if worker t is on duty in period p of day d
$s_{t,d,p}$	1 if worker t's shift starts in period p of day d
$b_{t,d,p}$	1 if worker t's meal break starts in period p of day d
Auxiliary variables	
$g^{-}_{d,p}$	Undercoverage in period p of day d
$g^{+}_{d,p}$	Overcoverage in period p of day d

[a] Corresponds to the 11-hour interday rest under the ten-minute discretisation.

$$\min \sum_{d\in D}\sum_{p\in P} g^{-}_{d,p}. \tag{3.1}$$

subject to:

$$\sum_{t\in T} y_{t,d,p} - n_{d,p} = g^{+}_{d,p} - g^{-}_{d,p} \qquad \forall d \in D,\ p \in P \tag{3.2}$$

$$x_{t,d} \le a_{t,d} \qquad \forall t \in T,\ d \in D \tag{3.3}$$

$$\sum_{p\in P} s_{t,d,p} = x_{t,d} \qquad \forall t \in T,\ d \in D \tag{3.4}$$

$$\sum_{p\in P} y_{t,d,p} = L x_{t,d} \qquad \forall t \in T,\ d \in D \tag{3.5}$$

$$\sum_{p\in P} b_{t,d,p} = x_{t,d} \qquad \forall t \in T,\ d \in D \tag{3.6}$$

$$y_{t,d,p} = \sum_{p'=\max(0,\,p-\sigma+1)}^{p} s_{t,d,p'} - \sum_{p'=\max(0,\,p-\beta+1)}^{p} b_{t,d,p'} \qquad \forall t \in T,\ d \in D,\ p \in P \tag{3.7}$$

$$b_{t,d,p} \le \sum_{p'=\max(0,\,p-c_{\max})}^{p-c_{\min}} s_{t,d,p'} \qquad \forall t \in T,\ d \in D,\ p \in P \tag{3.8}$$

$$\sum_{d\in D_w} x_{t,d} \le K \qquad \forall t \in T,\ w \in W \tag{3.9}$$

$$\sum_{d\in D_w} x_{t,d} \ge \min\Big(K,\ \sum_{d\in D_w} a_{t,d}\Big) \qquad \forall t \in T,\ w \in W \tag{3.10}$$

$$\sum_{d'=d}^{d+\Psi} x_{t,d'} \le \Psi \qquad \forall t \in T,\ d \in D^{\Psi} \tag{3.11}$$

$$y_{t,d,p} + y_{t,d-1,\,p+\delta} \le 1 \qquad \forall t \in T,\ d \in D^{+},\ p \in P_{\delta} \tag{3.12}$$

$$x_{t,d}, y_{t,d,p}, s_{t,d,p}, b_{t,d,p} \in \{0,1\}, \quad g^{-}_{d,p}, g^{+}_{d,p} \ge 0 \qquad \forall t \in T,\ d \in D,\ p \in P \tag{3.13}$$

The objective (3.1) minimises the total undercoverage of the requirement profile, measured in worker-periods; multiplying this quantity by the ten-minute period length converts it into the missing hours used as the planning criterion in Section 3.1. The coverage balance (3.2) decomposes the difference between scheduled coverage $\sum_t y_{t,d,p}$ and the requirement $n_{d,p}$ into non-negative over- and undercoverage. Because only undercoverage is penalised, the requirement profile acts as a target whose undercoverage is minimised subject to the available workforce and labour rules, rather than as a hard lower bound that coverage must always meet.

The remaining constraints encode the labour rules of Table 3.1. Constraint (3.3) prevents assignment on days when a worker is unavailable. Constraints (3.4)–(3.8) describe the internal struc-

ture of a single working day: each working day has one shift start (3.4) and one meal break (3.6), comprises L periods of effective work (3.5), and links the shift start and break to the period-level duty variable through (3.7), so that a worker is on duty in the σ periods following the start except during the β break periods. Constraint (3.8) allows the break to begin only after the minimum work and before the maximum continuous working time. Constraints (3.9) and (3.10) impose the weekly working-day requirement, with the lower bound reduced when a worker is not available on every day of the week. Constraint (3.11) limits the number of worked days in any window of $\Psi + 1$ consecutive days, and (3.12) enforces the minimum interday rest by forbidding work in the early periods of a day when the corresponding late periods of the previous day were worked. Equation (3.13) states the variable domains.

Limitations of deterministic requirement profiles. The deterministic MIP model provides a structured baseline for the organisation's current planning logic: given the nominal requirement profile and the available workforce, it generates a feasible schedule that minimises staffing shortfall relative to IDEAL. Its limitation lies elsewhere. The model does optimise this deterministic objective, but the objective is evaluated only against the nominal profile supplied to it.

In operational terms, undercoverage carries a direct meaning beyond the slack variable that represents it. A positive value of $g^{-}_{d,p}$ means that, in period p of day d, the schedule assigns fewer workers than the staffing level the organisation expected the post to require. These missing worker-periods are therefore interpreted as planned staffing gaps. Such gaps are undesirable in practice, since they may force manual adjustments or redistribution of work and place additional pressure on the workers on duty, leaving the post less able to meet the expected workload and degrading the service offered to customers.

This interpretation motivates the methodology developed in the next chapter. The deterministic baseline minimises missing hours relative to IDEAL, but IDEAL remains a single deterministic representation of expected staffing needs. The question taken up next is therefore whether modifying the requirement profile before optimisation can produce schedules with less missing coverage when demand is evaluated under uncertainty, while keeping the underlying MIP model fixed.

Chapter 4

Proposed Methodology

This chapter presents the methodological framework developed in this thesis to study the stochastic performance of schedules generated by a deterministic staff scheduling MIP model, and to examine whether demand-buffer robustification can improve that performance. The chapter describes the general methodology separately from the specific data and computational details reported in the remaining chapters.

Figure 4.1 provides a high-level overview of the proposed methodology. The framework is organised into three main stages: input and experimental setup, calibration, and final evaluation. The first stage defines the planning inputs used throughout the study. Historical demand observations are divided into training and testing subsets (Section 5.5) and are then scaled, as well as the nominal requirement profile, according to the Total Coverage Constrainedness (TCC) specification (Section 4.4) used in each experimental dimension. This produces a TCC-controlled requirement profile and corresponding scaled historical demand observations, while preserving the separation between training and testing data. Two demand-side objects recur and should be kept apart: the nominal requirement profile is the planning target the buffer reshapes and against which shortfall is measured, whereas the historical demand observations are the realised data used to estimate variability and generate scenarios.

The second stage corresponds to calibration: choosing the buffer's parameters, together called its configuration, by testing candidate configurations on training scenarios and keeping the one with the best simulated performance. To this end, the training subset is used to estimate demand variability, construct the buffered requirement profiles (Section 4.5), and generate the calibration scenarios against which the configurations are tested. Candidate buffer configurations are evaluated through a simulation–optimisation loop (Section 4.1): a buffered requirement profile is supplied to the deterministic MIP (Section 4.2), the resulting schedule is evaluated under calibration scenarios (Section 4.3), and the estimated calibration performance guides the search over buffer parameters (Section 4.6). The output of this stage is a selected buffer configuration, or set of configurations that produces an optimal schedule to be carried forward to final evaluation.

The third stage corresponds to out-of-sample evaluation. The testing subset is used to generate testing scenarios that were not used during buffer construction or calibration. The deterministic

baseline schedule and the selected buffered schedules are then evaluated under the same testing scenario set, using the metrics defined in Section 4.7. This final comparison provides the evidence used to assess whether demand-buffer robustification improves stochastic performance relative to the deterministic baseline.

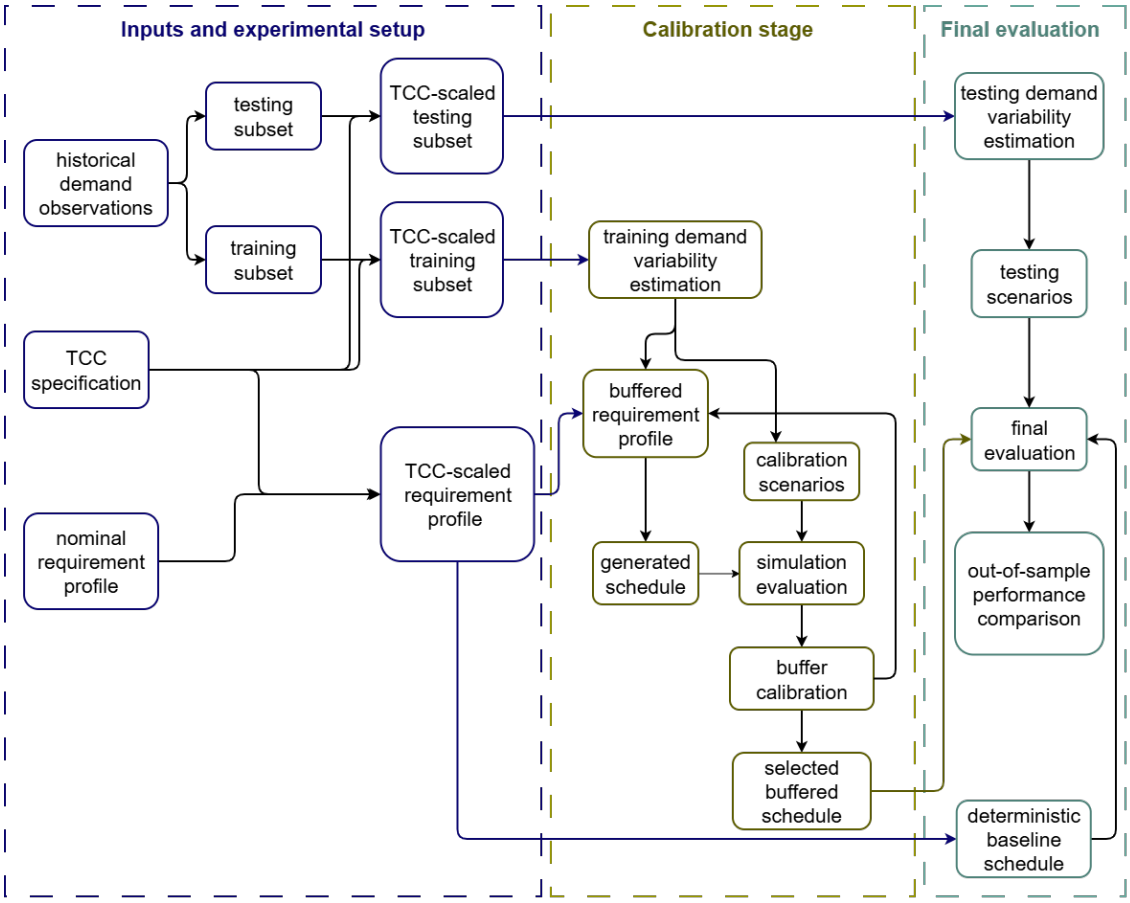

Figure 4.1: Overview of the proposed methodology. Historical demand observations are split into training and testing subsets. The training subset supports demand-variability estimation, buffer construction and parameter calibration, while the testing subset is reserved for the final out-of-sample evaluation of the deterministic baseline and selected buffered schedules. TCC is treated as an experimental dimension by consistently scaling the nominal requirement profile and the corresponding historical demand observations.

4.1 Simulation–optimisation framework

The methodology in this thesis brings together two computational components that are usually treated as alternatives (Figueira and Almada-Lobo, 2014): an optimisation procedure that produces candidate schedules, and a simulation procedure that evaluates how those schedules behave under conditions the optimisation procedure cannot represent on its own. Figueira and Almada-Lobo

(2014) classify simulation–optimisation methods according to several dimensions, including the purpose of the simulation component. In that dimension, Analytical Model Enhancement approaches use simulation to refine a parameter of an analytical optimisation model, so that the enhanced model itself yields solutions that perform well under uncertainty. The approach developed here belongs to this stream: the analytical model is the deterministic MIP scheduler, and the enhancement is a demand buffer, a calibrated margin added to the requirement profile the model receives. A set of buffer parameters is searched over a feasible domain; the deterministic MIP turns each candidate setting into a schedule; and simulation estimates that schedule's expected stochastic performance, which serves as the criterion guiding the calibration of the buffer. The simulation does not select the deployed schedule directly. Once the buffer is calibrated, it is a fixed enhancement, and the unchanged MIP produces the schedule from the buffered requirement, with no simulation required at deployment.

Optimisation generates feasible schedules. The deterministic MIP model takes a set of staffing requirements (the nominal profile, IDEAL, or a buffered version of it) and returns a schedule that satisfies the labour and operational constraints of the original model. This part of the framework is deterministic: given an input profile, it produces one schedule, optimised against that profile as if demand were certain.

Simulation is used to assess how the generated schedule performs once the certainty assumption is relaxed. Rather than rebuilding the optimisation model to represent uncertainty directly through scenario-based formulations or recourse structures, the framework leaves the structure of the deterministic MIP untouched, calibrating only the demand buffer applied to its input, and evaluates the resulting schedule against realised-demand scenarios constructed from the appropriate historical subset for the stage considered (calibration or final evaluation). Each scenario represents one possible demand realisation over the planning horizon, and for each one the gap between scheduled coverage and realised demand is recorded. Looking at this gap across multiple scenarios gives a stochastic performance profile for the schedule, something the deterministic model alone cannot offer.

The two components are coupled through the calibration loop shown in Figure 4.2. A candidate buffer configuration transforms the nominal requirements into a buffered profile. The deterministic MIP model then solves this modified instance and produces a schedule. This schedule is evaluated against a fixed set of training demand scenarios, producing an estimate of its stochastic performance. This estimate is the objective value used to compare buffer configurations during calibration. Grid search performs this comparison by enumerating a discretised parameter domain, whereas Bayesian optimisation builds a surrogate approximation of the relationship between the buffer configuration and stochastic performance and uses it to guide the search toward promising regions.

Once a buffer configuration is selected, its final performance is assessed separately on testing scenarios that were not used during calibration.

This framing fits the dissertation for three reasons. First, it keeps the deterministic MIP intact, since uncertainty enters upstream as a change to the input requirements rather than a reformula-

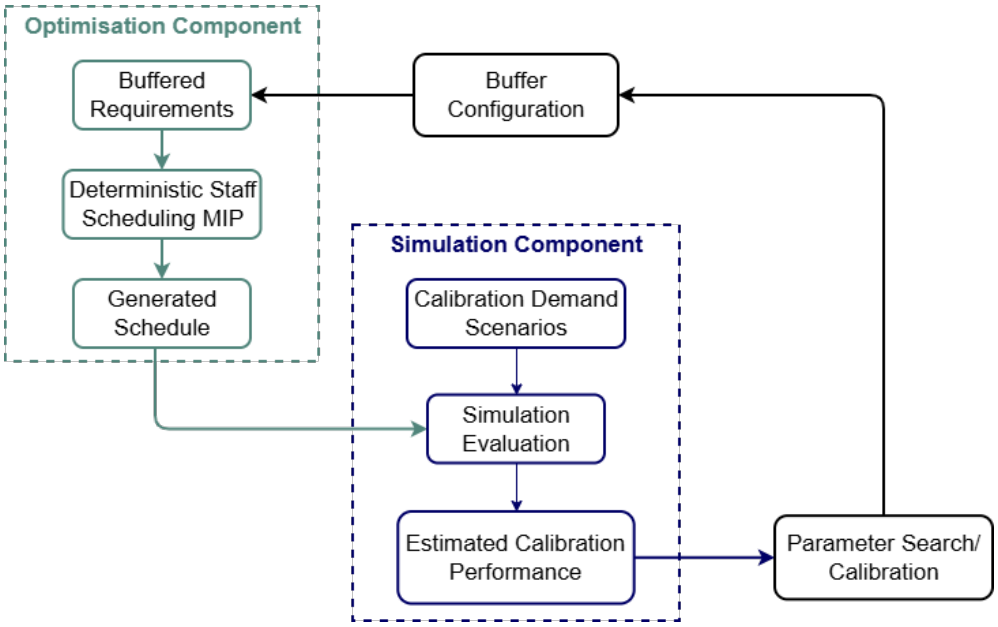

Figure 4.2: Simulation–optimisation calibration loop used to tune demand-buffer configurations. Each candidate buffer configuration is transformed into a buffered requirement profile, solved by the deterministic MIP, and evaluated on calibration scenarios generated from the training data. The estimated calibration performance guides the parameter search procedure.

tion of the model, matching the interest in a mechanism that sits on top of an existing scheduling tool rather than replacing it. Second, it enables uncertainty to be studied empirically through simulation, instead of the simplifications a tractable stochastic or robust reformulation would force, giving a common basis for comparing more than one uncertainty specification. Third, the same evaluation procedure applies to schedules built under different constrainedness levels and uncertainty specifications, which is what makes it possible to ask whether, and under what conditions, demand-buffer robustification changes stochastic performance.

This framework does not search for an optimal stochastic schedule: the deterministic MIP still optimises against one fixed input profile and simulation only evaluates the resulting schedule, offering a structured comparison of when and how demand-buffer robustification affects stochastic performance rather than a guarantee of any particular result.

4.2 Deterministic optimisation layer

This section describes the first of these two components: the deterministic optimisation layer that generates a schedule from a given staffing requirement profile.

The deterministic staff scheduling MIP model that forms this layer was presented in Chapter 3, along with its variables, constraints, objective function and operational context, and that formulation is not repeated here. For the purposes of the methodology, what matters is not the internal structure of the model, but the role it plays within the framework: it is treated as a fixed deterministic scheduling engine that maps a given requirement profile onto a schedule.

Given a staffing requirement profile R, the MIP model returns a schedule

$$x(R) \in \arg\min_{x \in \Omega} F(x, R), \tag{4.1}$$

where Ω is the feasible region defined by the labour and operational constraints described in Chapter 3, and $F(x,R)$ is the deterministic objective function, evaluated against the requirement profile R. Throughout this dissertation, $x(R)$ denotes the schedule obtained by solving the deterministic MIP model with requirement profile R, whether R is the nominal profile or a buffered one.

Two instances of this mapping recur throughout the methodology. The baseline schedule is obtained by solving the MIP model with the nominal staffing requirements, IDEAL:

$$x^0 = x(\text{IDEAL}). \tag{4.2}$$

Buffered schedules are obtained by solving the same MIP model with a modified requirement profile, $\widehat{R}(\theta)$, where θ denotes the parameters of the demand buffer applied — for example, a single conservatism level Δ that controls how far the requirement is raised above its nominal value (the global buffer), or a vector $\boldsymbol{\Delta}$ of such levels across temporal blocks (the block-specific buffer), both defined in Section 4.5:

$$x^\theta = x(\widehat{R}(\theta)). \tag{4.3}$$

In both cases, the resulting schedule determines the scheduled coverage $C_{d,p}(x)$: the number of workers scheduled in day d, period p, under schedule x. Scheduled coverage is the quantity against which the simulated demand is later compared, and it is defined in the same way for baseline and buffered schedules alike.

Two points about this layer are worth making explicit. First, the MIP model optimises x against whichever requirement profile R it receives, and treats that profile as fixed and known; it does not represent demand uncertainty, scenarios, or distributions of any kind. From the model's perspective, R is simply a deterministic input. Second, the same model is used for both the baseline and the buffered variants: the constraints, the feasible region Ω and the objective function F do not change between x^0 and x^θ. Only the requirement profile supplied as input changes.

This last point underlies how demand buffers are understood in this dissertation. Because each worker is constrained to work a fixed number of days and each working day has a fixed shift duration, the total scheduled capacity is constant across baseline and buffered schedules. Demand buffers therefore do not change the amount of capacity available or scheduled; they change the requirement profile that guides where this fixed capacity is allocated across the planning horizon. The deterministic optimisation layer is what turns a change in that signal into a change in the resulting schedule, while the scheduling problem itself, and the way it is solved, remain the same.

What this layer cannot do is say how well a given schedule, x^0 or x^θ, will hold up once actual demand differs from the profile it was optimised against. That is the question taken up in

Section 4.3.

4.3 Stochastic evaluation layer

This section describes the stochastic evaluation layer: how a generated schedule, baseline or buffered, is confronted with simulated demand scenarios, and what outputs this comparison produces.

Once a schedule x has been generated, whether x^0 or x^θ, it is evaluated against a scenario set S. Each scenario $s \in S$ specifies a realised demand $A^s_{d,p}$ of workers for every day d and period p, representing one plausible demand realisation over the planning horizon. The schedule itself fixes the scheduled coverage $C_{d,p}(x)$, the number of workers scheduled in day d, period p, as defined in Section 4.2. The evaluation layer works by comparing these two quantities, scheduled coverage against realised demand, scenario by scenario, day by day and period by period.

The primary scenario-level quantity considered is missing coverage, defined for scenario s, day d and period p as

$$M^s_{d,p}(x) = \max\{0, A^s_{d,p} - C_{d,p}(x)\}. \tag{4.4}$$

It records how far realised demand exceeds scheduled coverage in a given scenario, day and period: zero when scheduled coverage meets or exceeds realised demand, positive otherwise. Summed over all days and periods of the planning horizon, it gives the total missing coverage in scenario s,

$$M^s(x) = \sum_{d \in D} \sum_{p \in P} M^s_{d,p}(x). \tag{4.5}$$

$M^s_{d,p}(x)$ and $M^s(x)$ are the scenario-level quantities that the aggregated metrics used later in this dissertation are built from. Their full definitions are given in Section 4.7, although all of them trace back to the same comparison: realised demand against scheduled coverage, computed scenario by scenario.

The scenario set S depends on the stage of the methodology. During calibration, $S = S_{\text{cal}}$, and the scenarios are generated from the training subset. These scenarios provide the performance signal used to compare candidate buffer configurations. During final evaluation, $S = S_{\text{test}}$, and the scenarios are generated from the held-out testing subset. These testing scenarios are used only after the buffer configurations have been selected, so that final performance is assessed out of sample.

The stochastic evaluation layer does not depend on the specific scenario-generation procedure. It only requires a set of realised-demand scenarios $A^s_{d,p}$ against which each schedule can be evaluated. Chapter 5 specifies how the calibration and testing scenario sets are generated from the corresponding historical subsets.

It is important to be clear about what buffered schedules are evaluated against. A buffered schedule x^θ is produced by solving the deterministic MIP on a buffered profile $\widehat{R}(\theta)$ instead of IDEAL; but once x^θ exists, it is evaluated against the same realised-demand scenarios as the

baseline, not against the buffered profile that produced it. The buffer does its work upstream, at the optimisation stage, shaping which schedule gets generated.

Within each evaluation stage, the deterministic baseline and the buffered schedules need to be evaluated against the same scenario set. This corresponds to the use of common random numbers applied to this setting: if the realised-demand scenarios are held fixed across schedules, any performance difference that shows up can be attributed to the schedules themselves rather than to the simulated demand each one happened to be tested against.

What this layer ultimately does is to turn any schedule generated by the deterministic optimisation layer into a set of performance estimates. These estimates make it possible to compare the baseline against buffered schedules and also drive the search for buffer parameters described later in this chapter. The next section introduces TCC, a structural property of the scheduling instance that becomes one of the central experimental dimensions of the methodology.

4.4 Total Coverage Constrainedness as an experimental dimension

The previous sections established that schedules are generated by the deterministic MIP model and then evaluated under stochastic demand. Whether the buffer-based robustification mechanism improves that stochastic performance can depend partly on how buffers are defined and partly on a structural property of the instance: the extent to which the nominal requirement profile already absorbs the available workforce capacity. When that capacity is almost entirely consumed by nominal requirements, the solver has limited freedom to redistribute hours across periods. An instance in which nominal requirement is less tightly matched to available capacity gives the solver more room to respond to the redistributive signal introduced by a demand buffer. This degree of structural constrainedness is quantified by the TCC.

The use of TCC in this dissertation is motivated by the work of Ingels and Maenhout (2019), which is particularly relevant because it studies capacity-buffer allocation in personnel shift rosters and explicitly treats structural constrainedness as part of the experimental design. Their problem is not identical to the one addressed here: their model determines the size and position of capacity buffers endogenously, reformulating expected staffing requirements as stochastic requirements and allocating an uncertainty budget across shifts and days to define appropriate buffers, which creates a cost trade-off since capacity buffers may increase planned personnel presence, wage costs, overstaffing and cancellation-related costs. It nonetheless raises a closely related question: whether the effectiveness of a proactive buffer mechanism depends on how tightly staffing requirements consume the available workforce capacity. In their formulation, TCC is defined as the ratio between the imposed staffing requirements and the maximum theoretical coverage that the available employees can provide. For their artificial full-factorial test design, they generate expected staffing requirements at TCC values of 0.30, 0.40, and 0.50, while their real-life instances exhibit higher constrainedness, with TCC values of about 0.60 that indicate high employee utilisation and limited room for capacity buffers within the existing workforce. To examine whether this limitation reflects insufficient available capacity, they introduce additional anonymous employees in separate

staffing-size variants, which raises theoretical workforce capacity and brings TCC down to approximately 0.55 and 0.50, under which their proposed methodology performs more favourably. Their analysis therefore suggests that structural constrainedness affects the practical effectiveness of capacity-buffer strategies, although the mechanism through which this occurs differs from the fixed-capacity setting studied here. This motivates the adoption of TCC in the present dissertation as a controlled experimental dimension for analysing demand-buffer robustification under different levels of aggregate constrainedness.

In this dissertation, no workers are added. The workforce is fixed, each worker operates under the same contractual conditions (a fixed number of working days and a fixed shift duration per working day) and total scheduled capacity is constant across all schedules regardless of the buffer applied. TCC is therefore a structural property of the instance for any given workforce configuration. Different constrainedness levels are obtained not by modifying workforce capacity but by scaling the IDEAL profile: the aggregate level of nominal demand is adjusted relative to fixed available capacity while the temporal structure of demand is preserved. The same scaling factor is applied to the historical demand observations used to construct the stochastic demand scenarios, ensuring that each constrainedness level represents a coherent planning instance rather than only a modified optimisation input.

In this dissertation, TCC is defined as

$$\mathrm{TCC} = \frac{\sum_{d \in D} \sum_{p \in P} \tau_p \, \mathrm{n}_{d,p}}{Q}. \tag{4.6}$$

where τ_p is the duration of period p and Q is the total available workforce capacity over the planning horizon, expressed in the same units. Here, IDEAL denotes the nominal requirement profile of the instance under analysis. When the model uses worker-periods rather than hours, the equivalent ratio holds between total nominal requirements and total available capacity in worker-periods. A TCC value close to one means nominal demand nearly exhausts available capacity, leaving little room for reallocation. A lower value implies aggregate slack exists, though its distribution across periods depends on the structure of the requirement profile.

Whether this constrainedness effect arises in a fixed-capacity setting, and to what extent, is treated as an empirical question. The problem formulations differ substantially, and the findings of Ingels and Maenhout (2019) cannot be assumed to carry over directly. The aim is therefore to examine whether demand-buffer robustification remains sensitive to structural constrainedness when the mechanism operates through demand-side profile adjustment rather than endogenous capacity-buffer allocation.

Two constrainedness levels are examined in the computational study. These are controlled experimental dimensions, not a systematic sweep of the TCC range, and their selection is constrained by the characteristics of the real planning instance described in Chapter 3. TCC functions as an aggregate proxy for how tightly nominal demand consumes available capacity, and the two levels are chosen to represent materially different degrees of constrainedness within the bounds the real data allow.

The following section introduces the demand-buffer robustification strategy: the mechanism through which the requirement profile supplied to the MIP model is modified, and the two buffer specifications examined in the computational study.

4.5 Robustification mechanism

Building on the deterministic optimisation layer (Section 4.2), the stochastic evaluation layer (Section 4.3) and TCC (Section 4.4), this section introduces the mechanisms through which the deterministic scheduling process is prepared for, and then adjusted to, demand uncertainty.

Two mechanisms are considered. The first is an objective-level regularisation term that is applied uniformly to all schedules generated in the computational study. This term does not modify the requirement profile and is not calibrated as a separate robustification strategy. Its role is to regularise the deterministic MIP so that, when several schedules produce similar or identical levels of undercoverage, the selected solution does not concentrate surplus coverage excessively in a small number of periods. In this sense, it acts as a common preparatory mechanism that makes the subsequent demand-buffer experiments more comparable.

The second mechanism is the demand-buffer robustification strategy. This is the main calibrated mechanism studied in the dissertation. It modifies the requirement profile supplied to the deterministic MIP model before optimisation, while leaving the feasible region and labour constraints unchanged. The resulting buffered schedules are then evaluated under stochastic demand scenarios and compared with the regularised deterministic baseline.

4.5.1 Objective-level regularisation as a common preparatory mechanism

The original deterministic objective minimises total undercoverage relative to the requirement profile supplied to the MIP, in line with the organisation's planning criterion. When workforce capacity is fixed, however, several schedules can achieve the same total undercoverage while distributing surplus coverage differently across the horizon. This matters because the demand buffer introduced later also redistributes the fixed capacity: if the unbuffered objective lets surplus coverage concentrate in a few periods, a buffered profile may improve stochastic performance partly by forcing that surplus to be redistributed, rather than through the calibrated buffer itself. The regularisation term controls for this by applying the same mild preference for less concentrated overcoverage to the baseline and to all buffered schedules, making their comparison more uniform.

To reduce this source of arbitrariness, an auxiliary variable m is introduced to bound the maximum period-level overcoverage. Let $g^{-}_{d,p}$ and $g^{+}_{d,p}$ denote, respectively, the undercoverage and overcoverage in day d, period p, relative to the requirement profile being solved. The auxiliary variable satisfies

$$m \geq g^{+}_{d,p}, \qquad \forall d \in D,\ p \in P. \tag{4.7}$$

The deterministic objective is then regularised as

$$\min \sum_{d \in D} \sum_{p \in P} g^{-}_{d,p} + \lambda m, \tag{4.8}$$

where $\lambda > 0$ is a small weight. The first term remains the primary objective: the model still minimises total undercoverage relative to the requirement profile it receives. The second term acts only as a secondary regularisation component, discouraging solutions in which overcoverage is highly concentrated.

This mechanism does not add workforce capacity and does not change the feasible region of the deterministic MIP. It only affects the selection of a schedule within the feasible region by adding a secondary preference over otherwise comparable solutions. For this reason, it is not treated as an alternative robustification strategy in the experimental comparison. Instead, it is used as a common regularised version of the deterministic scheduling engine, applied both to the baseline and to all buffered schedules. Demand-buffer robustification is therefore evaluated on top of the same objective-level regularisation mechanism in all cases.

4.5.2 General buffer formulation and confidence-bound logic

For each day $d \in D$ and period $p \in P$, let $B_{d,p}(\theta) \geq 0$ denote the buffer term added to the nominal requirement $\mathrm{n}_{d,p}$, where θ denotes the buffer configuration. The buffered requirement profile is defined as

$$\widehat{R}_{d,p}(\theta) = \mathrm{n}_{d,p} + \mathrm{round}\left(B_{d,p}(\theta)\right), \quad \forall d \in D,\ p \in P. \tag{4.9}$$

where round$(\cdot)$ rounds to the nearest integer, since staffing requirements are expressed as integer headcounts, consistent with the scheduled coverage $C_{d,p}(x)$ defined in Section 4.2.

The restriction $B_{d,p}(\theta) \geq 0$ ensures that a buffered requirement is never lower than the corresponding nominal requirement, $\widehat{R}_{d,p}(\theta) \geq \mathrm{n}_{d,p}$ for all d, p. Setting $\theta = 0$ recovers the nominal profile,

$$\widehat{R}(0) = \mathrm{IDEAL}. \tag{4.10}$$

so that the deterministic baseline $x^0 = x(\mathrm{IDEAL})$, defined in (4.2), is the special case of $x^\theta = x(\widehat{R}(\theta))$ obtained at $\theta = 0$. $\widehat{R}(\theta)$ is not a realised-demand scenario and should not be interpreted as a guarantee that realised demand will not exceed it. It is a deterministic, conservative target supplied to the MIP, intended to reshape which periods the solver treats as tight and which it treats as slack, so that the reallocation of capacity described in Section 4.2 can anticipate demand variability rather than respond only to a single nominal value.

Within this general formulation, the buffer term $B_{d,p}(\theta)$ is specified through a deviation measure $D_{d,p}(\Delta)$, governed by a parameter $\Delta \geq 0$ that is referred to here as a conservatism parameter, or, where it admits such an interpretation, a confidence-level parameter. The deviation measure

$D_{d,p}(\Delta)$ represents a conservative deviation added above $n_{d,p}$, and it is required to satisfy three general properties: it is non-negative, it is equal to zero when $\Delta = 0$,

$$D_{d,p}(0) = 0, \quad \forall d \in D,\ p \in P. \tag{4.11}$$

– which, together with (4.9), reproduces $\widehat{R}(0) = \text{IDEAL}$ from (4.10) – and is non-decreasing in Δ, so larger values of Δ correspond to more conservative buffered requirements. Its insertion into $B_{d,p}(\theta)$ depends on the buffer specification considered, as defined next.

The choice to define $B_{d,p}(\theta)$ through a confidence-bound-inspired deviation measure, rather than, for instance, a deviation proportional to a fixed ratio of $n_{d,p}$, is motivated by the comparative analysis in Ingels and Maenhout (2019). In their personnel rostering setting, they compare a fixed ratio-based deviation measure with a confidence interval-based deviation measure, and find that the confidence interval-based measure, combined with an appropriate uncertainty budget allocation, achieves the best average actual performance among the strategies they test. This dissertation adapts that logic differently. In Ingels and Maenhout (2019), the confidence-bound logic is embedded in a model that determines the size and position of capacity buffers endogenously, as part of the rostering optimisation itself. Here, the confidence-bound idea is instead used as an exogenous transformation of the demand-side input supplied to the deterministic MIP: $D_{d,p}(\Delta)$ adjusts $n_{d,p}$ before the MIP is solved, the MIP itself is not reformulated, and $\widehat{R}(\theta)$ remains a buffered input profile.

The deviation measure $D_{d,p}(\Delta)$ is kept generic here because its precise functional form depends on the distributional approximation adopted for demand variability, which is a modelling choice of the computational study rather than part of the general methodology. It is instantiated in Chapter 5 from variability estimated on the training subset, with the testing subset reserved for final evaluation.

4.5.3 Global and block-specific buffer specifications

Two specifications of θ are examined in the computational study, corresponding to two granularities at which the conservatism parameter Δ can be applied across the planning horizon. Both are instances of the general formulation in (4.9); they differ only in how $B_{d,p}(\theta)$ is instantiated from the deviation measure and in how Δ is indexed.

Figure 4.3 illustrates the difference between the global and block-specific buffer specifications. In both cases, the buffer is applied before the deterministic MIP is solved, by transforming the scaled requirement profile into a buffered requirement profile. The figure is conceptual and does not represent a numerical result. Its purpose is to show that the buffer modifies the requirement signal supplied to the MIP, while the generated schedule remains constrained by the same fixed workforce capacity and scheduling rules.

The global and block-specific specifications formalise the two alternatives represented in Figure 4.3. The global buffer uses a single conservatism parameter for the full planning horizon,

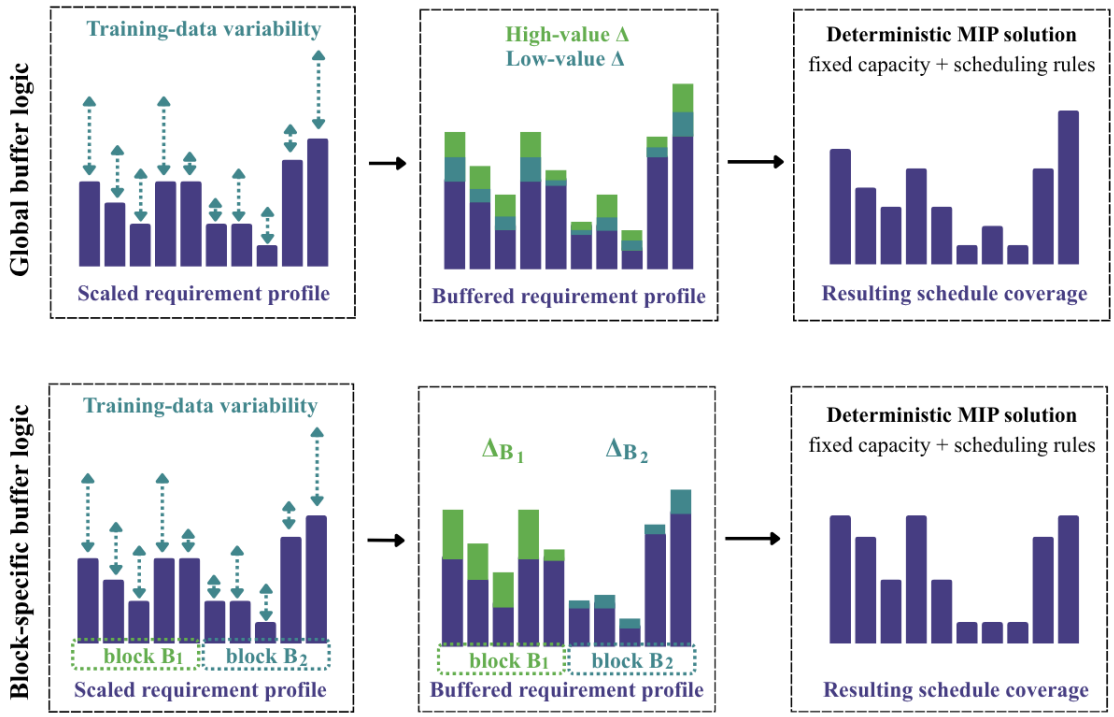

Figure 4.3: Conceptual illustration of the demand-buffer mechanism for the global and block-specific specifications. In both cases, the scaled requirement profile is adjusted using variability estimated from the training data before being supplied to the deterministic MIP. In the global specification, one conservatism parameter is applied across the horizon, although the resulting buffer margin may vary by period because variability differs. In the block-specific specification, different temporal blocks may receive different conservatism parameters. The MIP then generates scheduled coverage subject to fixed workforce capacity and scheduling rules, so the resulting coverage may follow the buffered requirement signal without necessarily reproducing it exactly.

whereas the block-specific buffer assigns separate conservatism parameters to predefined temporal blocks.

In the global buffer specification, a single conservatism parameter Δ is applied uniformly across the entire planning horizon. In this case θ reduces to the scalar Δ, and the buffered requirement profile becomes

$$\widehat{R}_{d,p}(\Delta) = \mathrm{n}_{d,p} + \mathrm{round}\left(D_{d,p}(\Delta)\right), \quad \forall d \in D,\ p \in P. \tag{4.12}$$

Although a single value of Δ is applied throughout the horizon, the resulting adjustment may not be the same across periods. Because $D_{d,p}(\Delta)$ may depend on the estimated variability of each period or time group, the buffer added to the nominal requirement can vary across days and periods even when the same value of Δ is used throughout the horizon. Figure 4.4 illustrates this mechanism: under a single conservatism level, the buffer added to a more variable period is larger than the one added to a stable period, and raising Δ widens both, by more where demand is more variable.

The global specification is simple and transparent, and it serves as a baseline robustification strategy against which finer-grained specifications can be compared. Its main limitation is that it cannot adapt the level of conservatism to particular temporal blocks: if some parts of the planning horizon would benefit from a different level of conservatism than others, a single global Δ cannot represent this.

The block-specific buffer specification addresses this limitation by partitioning the planning horizon into a set of temporal blocks $\mathscr{B}$, with a mapping $b(d,p) \in \mathscr{B}$ that assigns each day-period pair (d,p) to a block. A separate conservatism parameter Δ_b is associated with each block $b \in \mathscr{B}$, collected in the vector $\mathbf{\Delta} = (\Delta_b)_{b\in\mathscr{B}}$. In this case $\theta = \mathbf{\Delta}$, and the buffered requirement profile becomes

$$\widehat{R}_{d,p}(\mathbf{\Delta}) = \mathrm{n}_{d,p} + \mathrm{round}\left(D_{d,p}\left(\Delta_{b(d,p)}\right)\right), \quad \forall d \in D,\ p \in P. \tag{4.13}$$

The global buffer specification can be seen as the special case of (4.13) in which $\mathscr{B}$ contains a single block covering the entire planning horizon, so that $\Delta_{b(d,p)} = \Delta$ for all d,p. With more than one block, different blocks can receive different conservatism levels, allowing the level of conservatism to vary according to the temporal role or demand characteristics associated with each block. Whether this additional temporal granularity improves stochastic performance relative to the global specification, and under which constrainedness, is treated as an empirical question addressed in the computational study; the definition of the blocks themselves, including their number and boundaries, is also left to that chapter. In both specifications, the deterministic MIP model itself remains unchanged: only the requirement profile supplied to it, $\widehat{R}(\theta)$, differs from IDEAL.

The demand-buffer mechanism defines the family of buffered requirement profiles $\widehat{R}(\theta)$ that can be supplied to the deterministic MIP model in place of IDEAL, with θ ranging over a scalar Δ in the global specification or a vector $\mathbf{\Delta} = (\Delta_b)_{b\in\mathscr{B}}$ in the block-specific specification. The

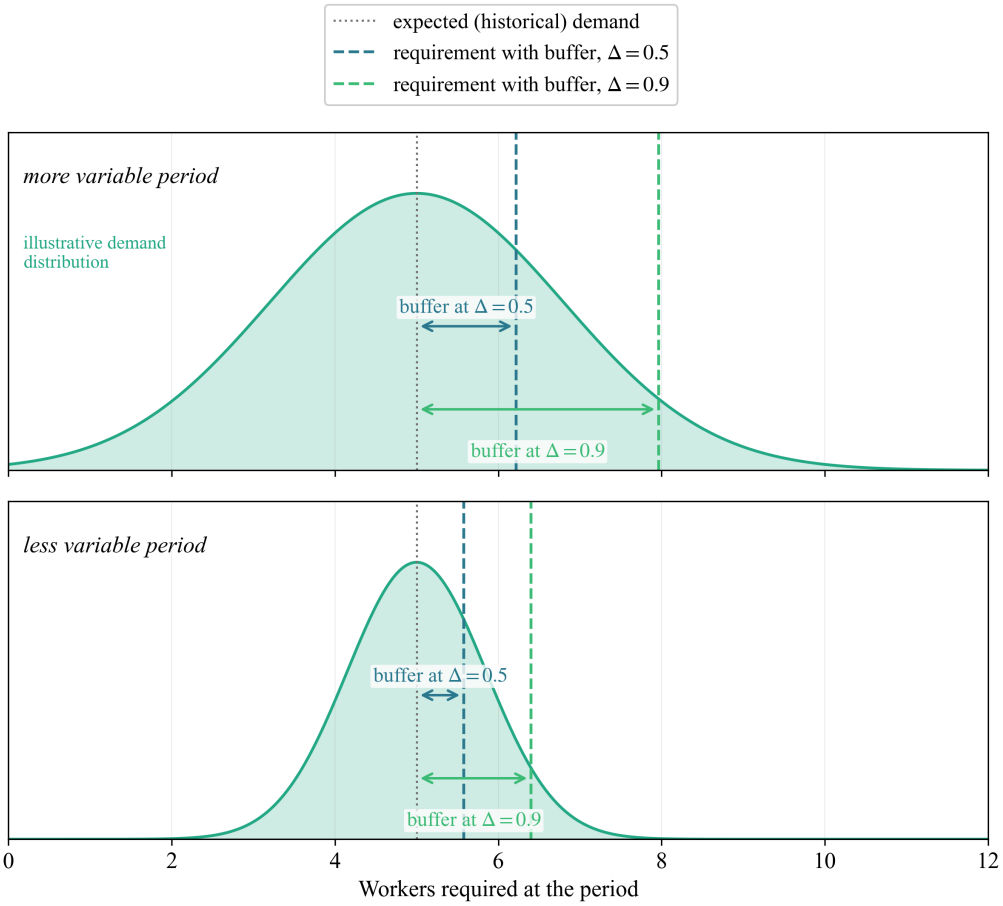

Figure 4.4: Conceptual illustration of how a single conservatism level Δ produces period-specific buffers. The buffer is the margin added above the period's nominal requirement to absorb demand variability; because the deviation measure $D_{d,p}(\Delta)$ scales with that variability, the same Δ yields a larger buffer in a more variable period than in a stable one, and raising Δ widens both. The demand distributions shown are illustrative; the specific distributional approximation used to instantiate $D_{d,p}(\Delta)$ is introduced in Chapter 5.

next question is how to select θ, whether scalar or vector-valued. This motivates the next section, which describes how θ is calibrated through simulation-based grid search and Bayesian optimisation, using the stochastic evaluation layer of Section 4.3 to guide the search over candidate buffer configurations.

4.6 Calibration of demand-buffer configurations

The two specifications of Section 4.5 leave θ unfixed. This section describes how θ is selected: given a candidate configuration, the corresponding buffered profile is constructed, the deterministic MIP generates the associated schedule, and the stochastic evaluation layer estimates that schedule's performance (Figure 4.2). This estimate allows candidate configurations to be compared and underlies the two calibration procedures described below, grid search and Bayesian optimisation.

4.6.1 Calibration objective

For a candidate configuration θ, let $x^\theta = x(\widehat{R}(\theta))$ denote the schedule generated by the deterministic MIP model from the buffered profile $\widehat{R}(\theta)$, as defined in Section 4.5. The calibration objective assigns to θ an estimate of the stochastic performance of x^θ, obtained by evaluating it against a set of calibration scenarios S_{cal}:

$$\widehat{J}(\theta) = \frac{1}{|S_{\text{cal}}|} \sum_{s \in S_{\text{cal}}} J\left(x^\theta, A^s\right). \tag{4.14}$$

where A^s denotes a realised-demand scenario, as introduced in Section 4.3, and $J(x^\theta, A^s)$ is a scenario-level performance criterion associated with schedule x^θ under scenario A^s. In this dissertation, the calibration criterion is based on the total missing coverage defined in Section 4.3. Equivalently, because all periods have the same duration, this criterion can be expressed as missing worker-time, by multiplying missing coverage by the ten-minute period length. Thus, $\widehat{J}(\theta)$ measures the average missing coverage of the schedule generated by configuration θ over the calibration scenario set S_{cal}.

Only this scalar criterion is used to guide grid search and Bayesian optimisation. The testing scenarios are not used at this stage, and the additional metrics defined in Section 4.7 are reserved for the final evaluation and interpretation of the selected schedules.

4.6.2 Grid search calibration

For the global buffer specification, θ reduces to a single scalar conservatism parameter Δ, as defined in Section 4.5. This low dimensionality makes it possible to calibrate Δ by grid search: a finite, predefined set of candidate values Δ_{grid} is specified, and $\widehat{J}(\Delta)$ is evaluated for each candidate in this set. The calibrated value is the one that minimises the calibration objective:

$$\Delta^{\star}_{\text{grid}} = \arg\min_{\Delta \in \Delta_{\text{grid}}} \widehat{J}(\Delta). \tag{4.15}$$

Because Δ_{grid} is enumerated exhaustively, grid search is transparent and easy to interpret in this scalar case: the performance of every candidate value is known, and the calibrated value can be compared directly against its neighbours in the grid.

This exhaustive approach is not extended to the block-specific buffer specification. There, the configuration is a vector $\boldsymbol{\Delta} = (\Delta_b)_{b \in \mathscr{B}}$, with one conservatism parameter per block. A grid covering all combinations of block-level values would grow combinatorially with the number of blocks, and each combination would require constructing a buffered requirement profile, solving the deterministic MIP, and running the stochastic evaluation layer. For more than a small number of blocks, this becomes computationally impractical, so grid search is not used to calibrate the block-specific buffer.

4.6.3 Bayesian optimisation calibration

Bayesian optimisation is used as a sequential calibration method for both buffer specifications, though it serves a different purpose in each. For the global buffer, it provides a sequential alternative to grid search over the scalar parameter Δ: it explores the same one-dimensional domain as Δ_{grid}, but without restricting evaluations to a fixed predefined set of points. Running both grid search and Bayesian optimisation on the global case lets the scalable method be validated against the exhaustive reference before it is relied upon for the block-specific buffers.

For the block-specific buffer, Bayesian optimisation searches over the vector $\boldsymbol{\Delta} = (\Delta_b)_{b \in \mathscr{B}}$, where it is particularly useful: as noted in Section 4.6.2, an exhaustive grid over $\boldsymbol{\Delta}$ would require evaluating all combinations of block-level values, a number that grows quickly with the number of blocks. Bayesian optimisation avoids this enumeration by selecting candidate configurations sequentially within a specified search domain.

In both cases, each candidate evaluation follows the same steps as in Section 4.6.1: the buffered profile $\widehat{R}(\theta)$ is constructed, the deterministic MIP is solved to obtain x^{θ}, and the stochastic evaluation layer is applied to estimate $\widehat{J}(\theta)$. Each evaluation requires solving the deterministic MIP and running the stochastic evaluation procedure, so only a limited number of configurations can be evaluated.

Bayesian optimisation makes the most of this limited budget by using the configurations evaluated so far to build a surrogate approximation of the relationship between θ and $\widehat{J}(\theta)$. The surrogate is updated as new evaluations are observed and guides the choice of the next candidate configuration, balancing exploration of regions of the search domain that remain uncertain against exploitation of regions where the surrogate suggests good performance is likely. The configuration ultimately selected is the one, among those evaluated, that achieves the lowest value of $\widehat{J}(\theta)$.

4.6.4 Selection of calibrated configurations

The role of each search strategy differs between the two buffer specifications. For the global buffer the search space is one-dimensional, so the exhaustive grid over Δ is the natural calibrator: it is inexpensive, it covers the tested range completely, and its result is directly interpretable. The grid is therefore used to specify the global buffer, selecting the value Δ^*_{grid} that minimises the calibration objective $\widehat{J}(\theta)$ over the grid. Bayesian optimisation is also applied to the global case, but as a diagnostic of the search method rather than as an alternative specification: because the grid provides an exhaustive reference in this low-dimensional setting, comparing the two shows whether the adaptive search reaches the same optimum, which in turn informs how far the Bayesian results can be trusted in the block-specific setting. For the block-specific buffer the search space is multi-dimensional and an exhaustive grid is infeasible (Section 4.6.2), so Bayesian optimisation is the calibrator: it returns the vector $\boldsymbol{\Delta} = (\Delta_b)_{b\in\mathscr{B}}$ that minimises $\widehat{J}(\theta)$.

Each selected configuration defines a corresponding buffered profile $\widehat{R}(\theta)$ and schedule x^θ (Sections 4.2 and 4.5). The configurations carried forward as buffer specifications are the global grid result and the block-specific Bayesian-optimisation result; the global Bayesian-optimisation result is reported alongside them as the search-method diagnostic.

Calibration and final evaluation are conceptually separate steps. The calibration objective $\widehat{J}(\theta)$ is computed on the calibration scenario set S_{cal}, and it is this objective that guides every selection above. The performance of the resulting configurations, including their comparison against the deterministic baseline x^0, is assessed separately on the test scenario set, using the evaluation metrics defined in the next section. Keeping calibration and final evaluation apart ensures that the choice of θ and the assessment of its consequences rest on different evidence.

4.7 Evaluation metrics and benchmark

Building on missing coverage and its scenario-level total $M^s(x)$ (Section 4.3), already used as the scalar calibration objective (Section 4.6.1), this section defines the broader set of metrics used to report and interpret the final performance of the deterministic baseline and the selected buffered schedules on the testing scenario set, and introduces a non-anticipative benchmark used as a reference point in that comparison.

Average missing coverage. The average missing coverage over a scenario set S is

$$\overline{M}_S(x) = \frac{1}{|S|}\sum_{s\in S} M^s(x). \tag{4.16}$$

This metric gives the average amount of uncovered demand that schedule x leaves across the scenarios in S. Depending on the stage of the analysis, S may denote the calibration scenario set or the final evaluation scenario set; the metric takes the same form in either case. Lower values of $M^s(x)$ indicate better average service performance.

Missing worker-time. Missing coverage can also be expressed in time units rather than headcount-periods. For a scenario s, the missing worker-time of schedule x is

$$H^s(x) = \sum_{d \in D} \sum_{p \in P} \tau_p M^s_{d,p}(x), \tag{4.17}$$

where τ_p is the duration of period p, and the corresponding average over S is

$$\overline{H}_S(x) = \frac{1}{|S|} \sum_{s \in S} H^s(x). \tag{4.18}$$

Expressed this way, a shortfall is reported in worker-hours rather than as a count of understaffed headcount-periods, which is a more direct measure of its size.

Tail-risk performance. The average metrics above capture central tendency, not how missing coverage varies from one scenario to another. The remaining metrics are not used to select buffer configurations. They are reported only in the final testing evaluation to characterise the schedules selected by the calibration stage. Two further metrics are based on the distribution of $\{M^s(x)\}_{s \in S}$. The first is the worst-case missing coverage,

$$M^{\max}_S(x) = \max_{s \in S} M^s(x), \tag{4.19}$$

and the second is a high quantile of the same distribution,

$$M^q_S(x) = Q_{q,S}\left(\{M^s(x)\}_{s \in S}\right). \tag{4.20}$$

where $Q_{q,S}$ denotes the q-quantile of the values $\{M^s(x)\}_{s \in S}$. These metrics show how a schedule performs in the scenarios with the highest shortfalls, which the average alone does not reveal.

Improvement relative to the deterministic baseline. To compare a buffered schedule x^θ with the deterministic baseline x^0, the improvement in average missing coverage is defined as

$$\text{Improvement}_S(x^\theta) = \frac{\overline{M}_S(x^0) - \overline{M}_S(x^\theta)}{\overline{M}_S(x^0)} \times 100. \tag{4.21}$$

A positive value indicates that x^θ reduces average missing coverage relative to x^0, while a negative value indicates that it performs worse. The same comparison can be applied to other metrics, such as missing worker-time, by substituting the corresponding quantity into the expression.

Service level. The metrics above measure the magnitude of the shortfall. A complementary operational measure is the service level, the share of (scenario, day, period) slots in which scheduled coverage meets realised demand:

$$\text{SL}_S(x) = \frac{1}{|S|\,|D|\,|P|} \sum_{s \in S} \sum_{d \in D} \sum_{p \in P} \mathbf{1}\left[A^s_{d,p} \le C_{d,p}(x)\right] \times 100. \tag{4.22}$$

Higher values indicate that demand is fully covered in a larger fraction of periods.

A non-anticipative benchmark. Alongside these metrics, the calibrated buffer is compared against a reference that bounds how much any fixed schedule can achieve. This benchmark replaces the requirement profile with the training demand distribution itself and computes, through a sample-average approximation, the single fixed roster that minimises expected missing coverage against it. Like the buffered schedules, it is non-anticipative, committing to one roster before demand is realised, but it uses different information, being fitted to realised historical demand rather than to the planned requirement. It is therefore a diagnostic, used to gauge how much room remains above the calibrated buffer, rather than a competing scheduling method. Its formulation is given in Appendix E and its results are discussed in Section 6.4.4.

Together, these metrics and the non-anticipative benchmark just described provide the basis for comparing the deterministic baseline, the global buffer and the block-specific buffer across the experimental dimensions considered in this dissertation. The methodological framework is now complete. Schedules are generated by the deterministic MIP, evaluated under realised-demand scenarios, robustified through demand-buffer configurations, compared using the missing-coverage measures defined above, and gauged against the benchmark. The next chapter applies this framework to the case-study data, describing the experimental design and calibration settings.

Chapter 5

Experimental Study Design

This chapter applies the methodology of Chapter 4 to the case-study data. Throughout the experiments, the deterministic MIP model is used as a fixed scheduling engine: it receives a requirement profile and returns a schedule that satisfies the labour and operational constraints, with its feasible region and objective structure unchanged. Demand buffers act only on the input requirement profile, so each experiment is a different instance of the same engine rather than a different model. This chapter describes how the operational problem is turned into experimental instances and how the demand environments are constructed. The generation of stochastic scenarios, the buffer calibration and the comparison protocol follow in later sections. Numerical results and their interpretation are reported in Chapter 6.

5.1 Experimental instance construction

The robustification study requires the MIP model to be solved many times, once for each requirement profile, planning window and constrainedness level, and again for every candidate buffer configuration examined during calibration. Solving the full operational instance at each of these points would be computationally impractical. The operational problem is therefore reduced to a set of smaller experimental instances that preserve the structure of the scheduling problem while keeping each solve tractable.

The experimental unit is a seven-day planning window. A week is short enough to make repeated solves feasible and long enough to retain the weekly structure of the problem, including daily demand variation, weekday and weekend differences, the fixed shift duration, break placement and the weekly working-day rule. To avoid drawing conclusions from a single week, the study uses six seven-day windows drawn from the historical record and repeats the same procedures over all of them for each constrainedness level.

The workforce is fixed at eleven workers in every instance. This is a reduction from the nineteen workers available in the reference operational setting, made for computational tractability: the number of binary variables in the MIP grows with the number of workers, days and periods,

and a smaller workforce allows each instance to be solved consistently across the many configurations the study requires. The requirement profile is scaled accordingly in Section 5.2, so that the reduced instance preserves the relationship between workforce capacity and nominal requirement. The workers are homogeneous and interchangeable and follow the same scheduling rule set, so a period is covered equally well by any available worker. This reduction changes the size of the instance, not the formulation of the MIP: the constraints, objective structure and feasible region are those of Chapter 3.

Two implementation choices require a brief note. First, the limit of six consecutive working days depends on work history before the window begins. A seven-day window solved in isolation would leave this rule artificially relaxed at the start of the horizon, because no prior working days would count towards the limit. A fixed initial-condition assumption on the pre-horizon work history is therefore included, so that the consecutive-day rule is active from the first day of each window. Its only role is to prevent this artificial relaxation; it is held fixed across all windows and conditions.

Second, all computational experiments use the objective-level regularisation mechanism introduced in Section 4.5.1. In the experimental implementation, the regularisation weight is set to $\lambda = 0.01$. This small value is chosen so that the regularisation term remains secondary to the main missing-coverage objective. At the scale of the analysed instances, it acts as a mild tie-breaking weight that discourages concentrated overcoverage without changing the role of total missing coverage as the primary optimisation criterion. The same value is applied uniformly to the deterministic baseline and to all buffered schedules.

5.2 Demand environments and TCC-controlled scaling

The implemented experiments use the original historical demand of the case-study post. The two conditions studied are constrainedness levels, introduced in Chapter 4. TCC measures how tightly the nominal requirement consumes the available workforce; a higher TCC therefore corresponds to a tighter, higher-pressure instance. Each constrainedness level, together with its scaled requirement profile and scaled historical demand, forms a demand environment.

Two targets are considered, $\mathrm{TCC} = 0.7$ and $\mathrm{TCC} = 0.5$. These conditions are created by scaling the requirement profile, not by changing the workforce: the MIP schedules eleven workers in both cases. A target level k is obtained with a scaling factor ϕ_k, with

$$\phi_{0.7} = \frac{11}{19}, \qquad \phi_{0.5} = \frac{11}{27}. \tag{5.1}$$

The denominator 19 corresponds to the reference operational setting. The denominator 27 is used only as a scaling reference to construct a lower–constrainedness level; it does not mean that twenty-seven workers are made available to the MIP, whose workforce stays fixed at eleven.

The scaled nominal requirement profile at target level k is

$$R^k_{d,p} = \max\{1,\ \text{round}(\phi_k\, \text{n}_{d,p})\}. \tag{5.2}$$

In the requirement profile considered, $\text{n}_{d,p} \geq 1$ for every period included in the planning horizon, so the lower bound of one worker does not create artificial demand in otherwise inactive periods; it only prevents an active period from being rounded down to zero after scaling.

The historical demand observations used to generate the stochastic scenarios are scaled consistently with the constrainedness level. These observations are integer counts, but scaling by ϕ_k can produce fractional values, so the realised demand is returned to integer headcounts with the ceiling function,

$$\widetilde{A}^k_{d,p} = \max\{0,\ \lceil \phi_k A_{d,p} \rceil\}. \tag{5.3}$$

The ceiling is used so that realised demand is not understated after scaling, which would otherwise make coverage appear easier than it is. Because both (5.2) and (5.3) involve rounding or ceiling operations, the realised constrainedness of a scaled environment is not exactly equal to its target. The values TCC $= 0.7$ and TCC $= 0.5$ should therefore be read as target constrainedness levels rather than exact realised values. Table 5.1 summarises the two implemented environments.

Table 5.1: Implemented demand environments.

TCC target	Scaling factor	Workers (MIP)	Requirement scaling	Demand scaling	Purpose
0.7	$11/19$	11	round, ≥ 1	ceiling	Reference, high constrainedness
0.5	$11/27$	11	round, ≥ 1	ceiling	Lower constrainedness

5.3 Historical demand characterisation

This section reports only the demand analysis that supports the experimental design; a full exploratory study is outside its scope. The historical demand at the post is markedly non-uniform. It rises and falls over the operating day and differs between weekday types, so demand at a given time on a weekday is not interchangeable with demand at the same time at the weekend. This temporal structure is the reason demand variability is modelled at a temporal level, rather than through a single aggregate distribution for the whole horizon. Figure 5.1 illustrates the historical demand profile against which the nominal requirement and the experimental scenarios are contextualised.

Demand variability is itself temporally heterogeneous: it differs across weekday types and across the day. Two granularities are used, for two distinct purposes. The normal-based buffer margins of Section 5.4 are based on the standard deviation estimated at the weekday and ten-minute-period level, $s^{\text{buf}}_{w(d),p}$, where $w(d)$ is the weekday type of day d and p is the ten-minute period; this matches the period-by-period discretisation at which the buffered requirement profile is applied. The stochastic scenarios of Section 5.5, by contrast, are sampled from empirical

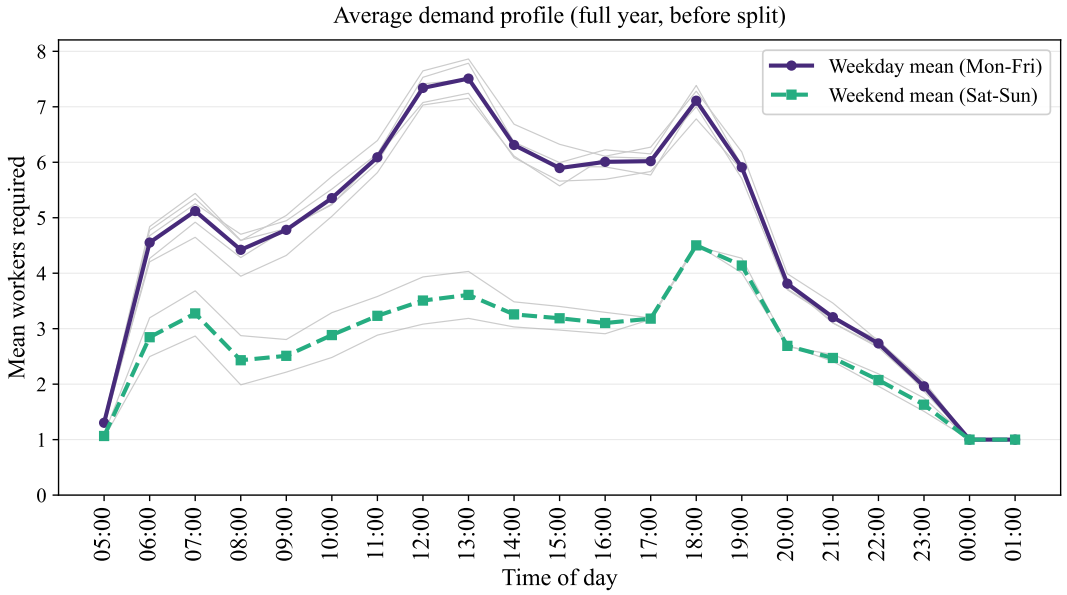

Figure 5.1: Average demand profile over the full historical year. Each line is the mean number of workers required per hour, averaged over weekdays (Mon–Fri) and weekend days (Sat–Sun); the faint grey lines show the individual weekday profiles.

distributions defined at the weekday and hour level. Figure 5.2 provides a compact hourly summary of demand variability in the training set. It is not the exact set of period-level values used to size the buffers; it shows that variability is temporally heterogeneous, which supports both the use of time-dependent uncertainty modelling and the weekday-hour grouping adopted for scenario generation.

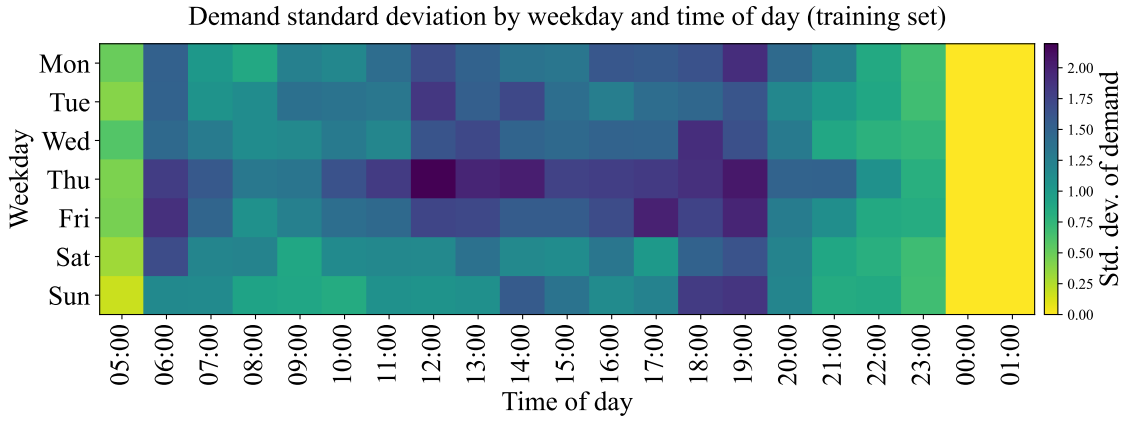

Figure 5.2: Hour-level standard deviation of demand by weekday and time of day, estimated on the training set. The figure provides a compact visual summary of temporal variability and supports the weekday-hour grouping used in scenario generation. Lighter cells indicate higher variability.

The one-year record is divided into a training and a testing half (Section 5.5). The split was checked with an outlier diagnostic based on the quartile rule: an observation is treated as an outlier

when it lies more than 1.5 times the interquartile range above the upper quartile of its weekday-hour group. The aim was not to show that the two halves are identically distributed, but to verify that the testing subset is not artificially easier than the training subset. Outlying observations are present in the testing subset across weekday types, and the two subsets are of comparable order; the split is not claimed to be perfectly balanced. The detailed outlier distributions by weekday and demand level are too dense for the main text and are reported in Appendix A. The same exploratory diagnostics also informed the temporal partition later used for the block-specific buffer calibration. In particular, the partition was designed to isolate a small number of temporally coherent regions with recurrent atypical demand behaviour, while keeping the number of block-level calibration parameters computationally manageable. The concrete partition is defined in Section 5.6.

5.4 Uncertainty approximation for buffer construction

The computational study uses the demand-buffer robustification mechanism defined in Chapter 4, but instantiates it with a normal-based approximation of demand variability. The deterministic MIP formulation is not modified. Instead, the buffer changes only the requirement profile supplied to the MIP.

For each TCC target k, the scaling described in Section 5.2 is applied before buffer construction. Thus, both the nominal requirement profile $R^k_{d,p}$ and the historical observations used to estimate demand variability are already expressed in the scaled demand environment associated with target k. Let $s^{\mathrm{buf},k}_{w(d),p}$ denote the sample standard deviation computed from the scaled training observations associated with weekday type $w(d)$ and ten-minute period p.

The normal approximation used for buffer construction treats the realised demand around the scaled nominal requirement as locally approximated by

$$A^k_{d,p} \approx \mathcal{N}\left(R^k_{d,p}, \left(s^{\mathrm{buf},k}_{w(d),p}\right)^2\right). \tag{5.4}$$

This approximation is used only to define an upper conservative requirement around the nominal target. It is not used to generate stochastic demand scenarios; those scenarios are generated empirically, as described in the following section.

For a global buffer with scalar conservatism parameter $\Delta \in [0,1)$, the corresponding standard-normal multiplier is

$$z_\Delta = \Phi^{-1}\left(\frac{1+\Delta}{2}\right), \tag{5.5}$$

where $\Phi^{-1}(\cdot)$ denotes the inverse cumulative distribution function of the standard normal distribution. The uncertainty margin is then

$$U^{k,\Delta}_{d,p} = \max\left\{0, z_\Delta s^{\mathrm{buf},k}_{w(d),p}\right\}. \tag{5.6}$$

The corresponding buffered requirement profile is

$$\widehat{R}_{d,p}^{k,\Delta} = \max\left\{1, \text{round}\left(R_{d,p}^{k} + U_{d,p}^{k,\Delta}\right)\right\}. \tag{5.7}$$

When $\Delta = 0$, $z_\Delta = 0$, so the margin is zero and the unbuffered scaled requirement is recovered:

$$\widehat{R}_{d,p}^{k,0} = R_{d,p}^{k}. \tag{5.8}$$

The final profile remains an integer staffing requirement. The workforce capacity is not increased; the modified requirement only changes the target against which the fixed-capacity MIP allocates workers.

The normal approximation was selected after exploratory graphical comparison with candidate distributions, including Normal, Weibull and Poisson alternatives. This comparison was not a formal goodness-of-fit test. The approximation is therefore used as a pragmatic modelling choice for constructing buffer margins, rather than as a complete probabilistic model of realised demand. Representative diagnostic plots are reported in Appendix B.

For block-specific buffers, the scalar Δ is replaced by a vector

$$\boldsymbol{\Delta} = (\Delta_b)_{b \in \mathscr{B}}, \tag{5.9}$$

where $\mathscr{B}$ denotes a set of temporal blocks. For a period p assigned to block $b(d,p)$, the corresponding uncertainty margin is

$$U_{d,p}^{k,\boldsymbol{\Delta}} = \max\left\{0, z_{\Delta_{b(d,p)}} s_{w(d),p}^{\text{buf},k}\right\}. \tag{5.10}$$

The corresponding buffered requirement profile $\widehat{R}_{d,p}^{k,\boldsymbol{\Delta}}$ is obtained exactly as in the global case (Equation (5.7)), with the block-specific margin $U_{d,p}^{k,\boldsymbol{\Delta}}$ replacing $U_{d,p}^{k,\Delta}$.

The temporal partition used for the block-specific buffer is introduced in the experimental protocol before the calibration settings.

5.5 Scenario generation and train–test split

The one-year historical record is divided temporally into two subsets. The last six months are used as the training and calibration subset, while the first six months are held out as the testing and final evaluation subset. This split is used consistently across the implemented demand environments. The training subset is used to estimate the variability terms required for buffer construction and to generate the calibration scenarios used during buffer parameter selection. The testing subset is used only for the final evaluation of the deterministic baseline and the calibrated buffered schedules. Because the two subsets are temporally disjoint, no observation used to estimate variability, construct buffers or select buffer parameters appears in the testing subset, so the out-of-sample

evaluation is free of data leakage. The diagnostic reported in Section 5.3 and Appendix A supports the use of the testing subset as a meaningful out-of-sample evaluation set.

Scenario generation is performed separately for each TCC target k. The scaled historical observations are defined in Equation (5.3). Scenarios are generated from these unbuffered scaled historical observations. Buffered requirement profiles are not used to generate realised-demand scenarios. Therefore, for a fixed planning window and demand environment, the realised–demand scenarios are the same when comparing the deterministic baseline and the buffered schedules.

Let $w(d)$ denote the weekday type of day d, p the ten-minute period, and $h(p)$ the hour to which period p belongs. For each scenario s, day d, and period p, realised demand is sampled from the empirical distribution of scaled historical observations with the same weekday type and hour:

$$A^{s,k}_{d,p} \sim \widehat{F}^{k}_{w(d),h(p)}. \tag{5.11}$$

The resulting scenario is applied at ten-minute-period resolution: each period p receives a realised demand value. The empirical sampling, however, is defined at weekday-hour level rather than at exact ten-minute-period level. This provides a larger empirical sample per weekday-hour group while preserving the main weekday and time-of-day structure of demand.

This empirical resampling procedure preserves weekday-hour marginal behaviour, but it does not model serial dependence or within-hour structure, which is a deliberate scope choice, since the objective is to create comparable stochastic demand realisations for evaluating schedules, rather than to estimate a full generative model of demand.

For each planning window and demand environment, 500 scenarios are generated using a fixed random seed. The number of scenarios was chosen to make the Monte Carlo error of the reported estimates negligible. Each scenario is an independent draw from the fitted empirical demand distribution, so the standard error of the estimated mean missing hours decreases as $1/\sqrt{|S|}$. At $|S| = 500$ it is below 0.1 hours, with a 95% confidence half-width below 0.2 hours, an order of magnitude smaller than the per-window improvements interpreted in Chapter 6; increasing the number of scenarios reduces only this Monte Carlo noise, not the information content, which is fixed by the historical sample underlying the empirical distribution.

The same scenario set is used to evaluate the deterministic baseline and all buffered schedules. This common-random-numbers design ensures that observed performance differences are attributable to the schedule and buffer configuration, rather than to scenario sampling noise.

5.6 Buffer configuration and calibration settings

This section specifies how the buffer mechanisms defined in Chapter 4 are instantiated in the computational study. It states the calibration settings of the global buffer, the temporal partition used by the block-specific buffer, the Bayesian optimisation settings shared by both, and the calibration effort associated with each buffer configuration. Calibration follows the general procedure of Section 4.6, constructs buffered requirement profiles as in Section 5.4, and uses the calibration and

testing scenario sets of Section 5.5. It is performed separately for each planning window and TCC target, so the settings below apply within a single (window, TCC) pair and are repeated across the experimental matrix.

Global buffer: grid search. Grid search is applied only to the global buffer, whose configuration is the single scalar conservatism parameter Δ. The grid contains ten values in the domain $[0,1)$,

$$\Delta_{\text{grid}} = \{0,\ 0.15,\ 0.30,\ 0.45,\ 0.60,\ 0.75,\ 0.90,\ 0.95,\ 0.975,\ 0.99\}.$$

The value $\Delta = 1$ is not included because, under the normal-based buffer formula of Section 5.4, it would require the quantile $\Phi^{-1}(1)$, which is not finite. For each value in the grid, a buffered requirement profile is constructed, solved by the deterministic MIP, and evaluated on the calibration scenarios. The selected value is the grid candidate with the lowest calibration objective.

Global buffer: Bayesian optimisation. Bayesian optimisation is also applied to the global buffer over the scalar Δ in $[0,1)$, as a sequential alternative to grid search. It is implemented with `gp_minimize` from `scikit-optimize` version 0.10.2, using a Gaussian process surrogate and the default random initial-point generator (`initial_point_generator='random'`), which samples points uniformly in the search domain. The random state is fixed at `random_state=42`. The procedure uses 30 evaluations per (window, TCC) pair: 10 initial random evaluations, followed by 20 sequential Bayesian optimisation iterations. The search stops at this fixed evaluation budget rather than on a convergence tolerance, and the selected value is the evaluated candidate with the lowest calibration objective.

Temporal partition for the block-specific buffer. The block-specific buffer assigns one conservatism parameter Δ_b to each block of a temporal partition $\mathscr{B}$, so the choice of partition fixes the dimensionality of the search vector $\boldsymbol{\Delta} = (\Delta_b)_{b\in\mathscr{B}}$. The partition used here contains five blocks. Four of them isolate weekday–hour regions where the exploratory outlier diagnostics of Section 5.3 pointed to atypical demand behaviour: B_1 covers Friday, Saturday and Sunday between 14:00h and 17:59h; B_2 covers Tuesday and Wednesday between 07:00 and 10:59; B_3 covers Monday between 20:00h and 23:59h; and B_4 covers Thursday between 20:00h and 23:59h. The fifth block, B_5, holds all remaining day-period combinations.

The partition was informed by these diagnostics, derived pragmatically from these diagnostics rather than from a formal clustering procedure. It isolates a limited number of regions where atypical demand appeared relevant, and several periods that also show outliers remain outside the four selected blocks and fall into the residual block B_5. The aim is not to classify every outlier-prone period, but to build a compact experimental partition for testing differentiated buffer levels. The result balances the outlier evidence against temporal interpretability, operational coherence and computational tractability.

The granularity is deliberately intermediate. Blocks that were too broad would leave the block-specific buffer unable to distinguish different temporal behaviours, so that it would behave much

like the global buffer. A partition that was too fine would raise the number of block-level parameters, since each additional block adds one component to $\boldsymbol{\Delta}$. Even though Bayesian optimisation explores the search space more efficiently than a full grid, a higher–dimensional vector requires more candidate evaluations to be explored adequately within the available budget. The five-block partition is therefore a compromise between temporal specificity and computational tractability.

The selected blocks also admit an operational reading, since they span different times of day: a morning region (Tuesday and Wednesday, 07:00h to 10:59h), an afternoon region (Friday, Saturday and Sunday, 14:00h to 17:59h), and two evening regions (Monday and Thursday, 20:00h to 23:59h), which are kept separate. Each selected block covers about four hours, roughly half of a worker shift, so the differentiated conservatism levels apply to meaningful portions of a shift rather than to isolated short intervals. Some blocks aggregate related days, while others are kept day-specific where a more localised calibration is useful.

Figure 5.3 and Table 5.2 describe the partition. The figure is shown at hourly resolution for readability, whereas the MIP operates at ten-minute-period resolution. In the implementation, each highlighted hour corresponds to all ten-minute periods it contains, and every period not assigned to B_1 through B_4 belongs to B_5.

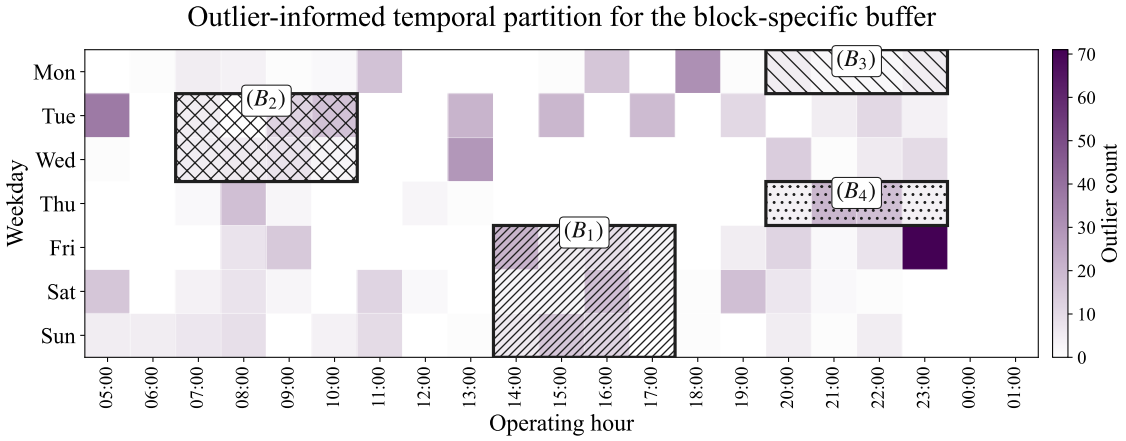

Figure 5.3: Temporal partition used for the block-specific buffer. Highlighted weekday-hour regions correspond to blocks B_1 to B_4, while all remaining periods are assigned to B_5.

Table 5.2: Temporal blocks used in the block-specific buffer partition.

Block	Days	Time interval (h)
B_1	Friday, Saturday, Sunday	14:00 to 17:59
B_2	Tuesday, Wednesday	07:00 to 10:59
B_3	Monday	20:00 to 23:59
B_4	Thursday	20:00 to 23:59
B_5	All other days	All remaining periods

Block-specific buffer: Bayesian optimisation. The block-specific buffer is configured by the vector defined in Equation (5.9) with one parameter $\Delta_b \in [0,1)$ per block. It is calibrated by

Bayesian optimisation only, under the same implementation, surrogate, initial-point generator and random state as the global buffer.

Grid search is not used here, since the number of block-level combinations grows combinatorially with the number of blocks. In the implemented design, the block-specific specification will use five temporal blocks. If the same ten-value grid used for the global buffer were applied independently to each block, a full factorial grid would require

$$|\Delta_{\text{grid}}|^{|\mathscr{B}|} = 10^5 = 100{,}000$$

candidate configurations for each $(\text{window}, \text{TCC})$ pair. Across the six planning windows and two TCC targets, this would amount to 1,200,000 calibration runs, each requiring a deterministic MIP solve and stochastic evaluation on the calibration scenarios. This computational burden motivates the use of Bayesian optimisation for the block-specific buffer. The procedure uses 120 evaluations per (window,TCC) pair: 50 initial random evaluations, followed by 70 sequential Bayesian optimisation iterations.

Selected configurations and calibration effort. For each planning window and TCC target, calibration produces three buffered configurations carried forward to final evaluation: the global buffer from grid search, the global buffer from Bayesian optimisation, and, once $\mathscr{B}$ is fixed, the block-specific buffer from Bayesian optimisation. In all cases, the calibration criterion is the missing-coverage performance obtained on the training scenarios: the selected configuration is the one that produces the lowest average missing coverage over S_{cal}.

Table 5.3 summarises the search settings of each configuration, and Table 5.4 reports the resulting number of calibration runs across the experimental matrix.

Table 5.3: Buffer calibration settings.

Buffer configuration	Calibration method	Search variables	Evaluations per (window, TCC)
Global buffer	Grid search	$\Delta \in \Delta_{\text{grid}}$	10 grid values
Global buffer	Bayesian optimisation	$\Delta \in [0, 1)$	30 (10 random + 20 sequential)
Block-specific buffer	Bayesian optimisation	$\boldsymbol{\Delta} = (\Delta_b)_{b \in \mathscr{B}}$, $\Delta_b \in [0, 1)$	120 (50 random + 70 sequential)

Table 5.4: Number of calibration runs per method.

Method	Windows	TCC targets	Runs per (window, TCC)	Total runs
Grid search (global)	6	2	10	120
Bayesian optimisation (global)	6	2	30	360
Bayesian optimisation (block-specific)	6	2	120	1440
Total				1920

These totals refer only to calibration evaluations. Additional solves are required to generate the deterministic baseline and to evaluate the selected calibrated configurations on the held-out testing scenarios.

The solver and implementation settings under which these runs are executed are reported in Section 5.7.

5.7 Computational settings and experimental matrix

This section summarises the computational settings of the implemented study and the experimental matrix that follows from them. The experimental unit is the combination of a planning window and a TCC target. Within each such combination, the deterministic baseline and the calibrated buffered schedules are evaluated under common scenario sets.

The experimental settings are summarised in Table 5.5: six seven-day windows (Section 5.1), a fixed workforce of eleven workers, the two TCC targets 0.7 and 0.5 (Section 5.2), and the original historical demand of the case-study post.

For each planning window, TCC environment and data subset, 500 realised-demand scenarios are generated using a fixed random seed. The same calibration scenario set is used to compare candidate buffer configurations during parameter selection, and the same testing scenario set is used to compare the deterministic baseline and the calibrated buffered schedules during final evaluation. The deterministic MIP is solved with Gurobi, and the surrounding instance construction, scenario generation and calibration procedures are implemented in Python.

The calibration methods are run under per-solve time limits. Grid search and the global Bayesian optimisation use a limit of 7200 seconds per solve. The block-specific Bayesian optimisation requires many more solves (Table 5.4), so a reduced limit of 900 seconds per solve is used to keep its calibration computationally tractable. The adequacy of this shorter limit is assessed in Section 6.4.5. The target optimality gap is MIPGap $= 0$, so the solver attempts to prove optimality, but the time limit may stop a solve before optimality is established. When a time limit is reached and a feasible incumbent solution exists, that incumbent and its reported solver gap are retained for analysis. All solves are single-threaded with a fixed seed, so that solves compared within each calibration method are run under identical conditions.

Table 5.5 summarises the main computational settings.

Table 5.5: Main experimental settings.

Dimension	Implemented setting
Planning horizon	7-day windows
Number of windows	6
Workforce	11 workers, fixed across all instances
TCC targets	0.7 and 0.5
Demand data version	Original historical demand
Scenarios per evaluation	500
Random seed	Fixed at 0 for Gurobi and 42 for the scenarios generation
Solver	Gurobi
Implementation language	Python
Grid search applicability	Global buffer only
Bayesian optimisation applicability	Global and block-specific buffers
Solve time limits	7200 s per solve (grid and global Bayesian); 900 s per solve (block-specific Bayesian)
Optimality gap target	MIPGap = 0, subject to time limits

Chapter 6

Experimental Results and Discussion

This chapter applies the methodology of Chapters 4 and 5 to the case-study data, to answer whether demand-buffer robustification improves the stochastic performance of schedules produced by the deterministic MIP, and under what conditions. The analysis proceeds in two steps: it first characterises the deterministic baseline under stochastic demand, and then assesses whether the global and block-specific buffers reduce the resulting shortfall. All quantities are reported out of sample at the two constrainedness levels $k = 0.7$ and $k = 0.5$, a tightly and a loosely loaded workforce. Section 6.1 establishes the deterministic baseline and its stochastic benchmark; Section 6.2 reports the global buffer; Section 6.3 the block-specific buffer; and Section 6.4 compares performance across constrainedness levels and specifications, explaining the mechanism and its limits.

6.1 Baseline behaviour and stochastic benchmark

This section describes the deterministic baseline schedule $x^{k,0} = x(R^k)$ before any demand buffer is applied. It first shows how the scheduling engine allocates the fixed workforce against the nominal requirement, with no uncertainty involved, and then measures how that schedule performs once demand is stochastic. All stochastic evaluation uses the held-out test scenarios S_{test}, and every quantity is reported for the two constrainedness levels $k = 0.7$ and $k = 0.5$.

6.1.1 Deterministic schedule behaviour

Figure 6.1 shows, for a representative planning window, the scheduled coverage $C_{d,p}(x^{k,0})$ together with the scaled requirement profile $R^k_{d,p}$ that the MIP optimised against, averaged over the seven days of the window. This is the model's own view of the solution, with no stochastic demand: the input is the requirement profile, not realised demand. The figure is therefore illustrative of the scheduling engine, not a performance result.

Because the workforce is fixed and constrained by the labour rules of the original model, the schedule cannot match the requirement at every period. Coverage tracks the daily requirement profile but leaves some residual undercoverage at the busiest periods, and some overcoverage where the labour rules keep workers on shift after the requirement is already met. The regularisation term

m (Section 5.6, $\lambda = 0.01$) keeps this overcoverage from concentrating in a few periods, so the baseline is a clean and reproducible reference for the buffered schedules. All baseline and buffered schedules are obtained under the same computational regime defined in Section 5.7, ensuring that comparisons are made under consistent solver conditions.

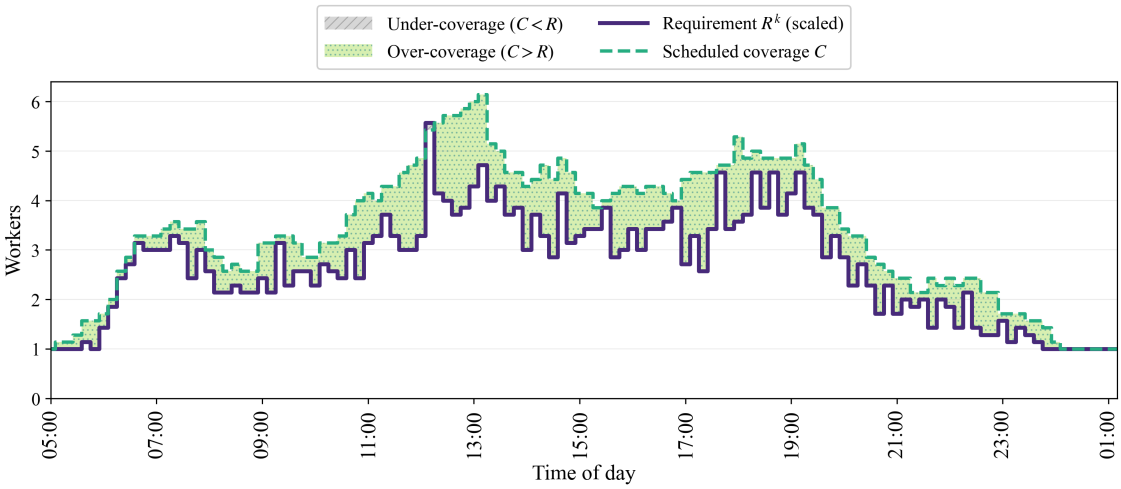

Figure 6.1: Illustrative deterministic behaviour of the baseline schedule for a representative window ($k = 0.7$): scheduled coverage C against the scaled requirement R^k, averaged over the seven days. No stochastic demand is involved; the input is the requirement profile. Shaded areas mark residual undercoverage ($C < R$) and overcoverage ($C > R$) under the fixed workforce.

6.1.2 Stochastic baseline performance

The baseline schedules are then evaluated against the test scenarios S_{test}, drawn from the held-out first half-year demand distribution. For each scenario s, the total missing coverage of Eq. (4.5) is expressed as missing worker-time (missing hours).

Figure 6.2 places the same schedule against this realised demand. Whereas Figure 6.1 compared coverage with the requirement it was built on, here it is set against the realised demand and its variability. Coverage still follows the daily profile, but realised demand, and in particular its upper tail, rises above coverage across many more periods than the requirement did. This is where the stochastic shortfall builds up.

Table 6.1 reports, per constrainedness level and averaged over the six planning windows, the deterministic shortfall against the requirement and the stochastic shortfall on the test scenarios: the mean missing hours, the 95th percentile (a tail-risk measure), the worst case (the maximum over scenarios), and the service level, which is the share of day–period slots in which coverage meets realised demand.

The contrast between the two blocks of Table 6.1 is the key point. Against the requirement it was built for, the baseline leaves almost no shortfall: essentially zero hours on average at $k = 0.5$ (zero in most windows; see Appendix C) and 2.5 hours at the tighter $k = 0.7$, which reflects a real capacity limit rather than a modelling problem. The scheduling engine therefore does what

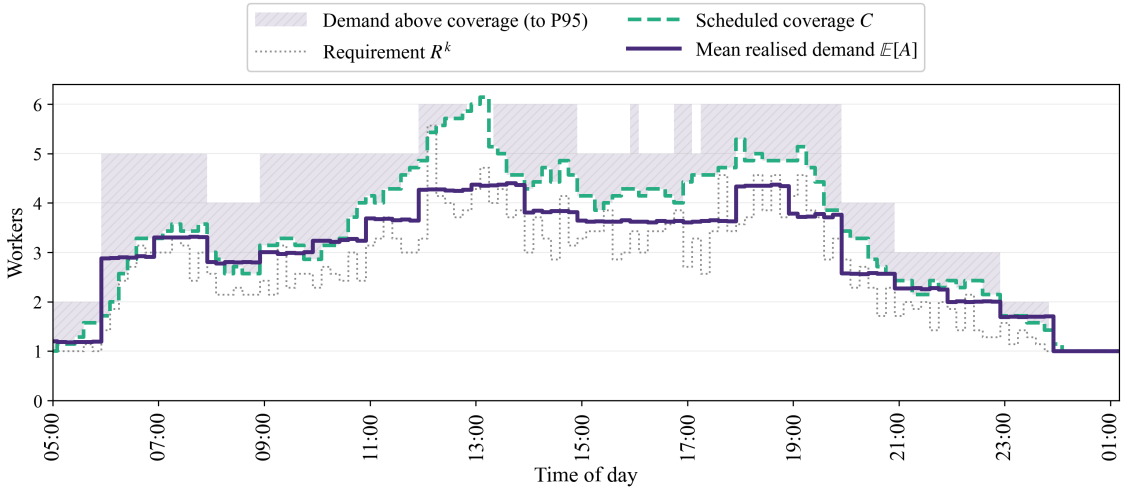

Figure 6.2: Baseline schedule under stochastic demand, for the same window as Figure 6.1 ($k = 0.7$). Scheduled coverage C (dashed) against the mean realised demand $\mathbb{E}[A]$ (solid), with the deterministic requirement R^k (thin, dotted) for reference. The shaded region marks where realised demand exceeds coverage, up to its 95th percentile.

Table 6.1: Baseline performance: deterministic shortfall against the requirement, and out-of-sample stochastic shortfall on S_{test}, by constrainedness level. All entries are means over the six planning windows; the worst case is the mean of the per-window worst cases, not a single worst scenario across all windows.

	TCC 0.7	TCC 0.5
Deterministic (schedule vs requirement R^k)		
Missing hours	2.5	0.0
Stochastic (test scenarios S_{test})		
Mean missing hours	37.9	9.1
P95 missing hours	41.8	10.9
Worst-case missing hours	44.9	12.5
Service level (%)	80.0	94.3

it should: it covers the nominal requirement almost entirely. The much larger stochastic shortfall, 9.1 and 37.9 missing hours, does not come from a weak optimiser but from demand uncertainty, since realised demand and its tail depart from the nominal requirement the schedule was built on. The baseline also produces overcoverage in several parts of the day, meaning that part of the stochastic shortfall is related to the temporal placement of coverage rather than only to the total amount of workforce capacity. This is the gap the demand buffer targets, and it is why the rest of the chapter acts on the requirement profile given to the MIP rather than on the MIP itself.

The two environments behave very differently. At the tighter level $k = 0.7$ the fixed workforce is heavily loaded relative to demand, and the baseline leaves a large expected shortfall (37.9 missing hours, service level 80.0%). At the looser level $k = 0.5$ the same workforce has more slack and already absorbs most of the demand (9.1 missing hours, service level 94.3%). This difference, close to fourfold in expected shortfall, is the main factor separating the two settings, since it leaves very different room for a buffer to act, and it frames the results of Sections 6.2 and 6.3. The test distribution is, if anything, slightly heavier than the calibration distribution (Section 5.5, Appendix A), so these figures are not optimistic.

As Figure 6.2 shows, coverage follows the mean realised demand closely, so the shortfall implied by the mean profile alone is small; yet the expected missing hours in Table 6.1 are large. The two facts fit together because the per-period missing function $\max\{0, A - C\}$ is convex in demand. By Jensen's inequality,

$$\mathbb{E}[\max\{0, A - C\}] \;\geq\; \max\{0, \mathbb{E}[A] - C\},$$

demand variability adds expected shortfall even in periods where the mean demand does not exceed coverage. The expected missing hours are driven not by a mismatch between the schedule and the average demand, which is small, but by the variability around it, visible as the band rising above coverage in Figure 6.2.

6.2 Global demand buffer results

This section reports the global demand buffer, in which a single scalar Δ is applied across the whole horizon. Section 6.2.1 describes how the buffer is calibrated and how grid search compares with Bayesian optimisation; Section 6.2.2 measures the calibrated buffer out of sample against the baseline.

6.2.1 Global demand buffer calibration

The global buffer is calibrated by grid search over Δ, with the calibration objective equal to the mean missing hours over the training scenarios. For each window and constrainedness level, the selected Δ is the grid value that minimises this objective. The calibrated conservatism differs sharply between the two environments: at $k = 0.7$ it settles on moderate values (median $\Delta = 0.75$),

and at $k = 0.5$ it settles near the top of the grid (median $\Delta \approx 0.98$). The per-window values are listed in Appendix C, and the interpretation of this difference is taken up in Section 6.4.

Figure 6.3 shows the calibration objective as a function of Δ for all six windows, together with their mean. The two levels respond very differently to the buffer, and because the panels are on different vertical scales the comparison is best made in relative terms. At $k = 0.5$ the mean objective falls by about 46% from its no-buffer value, reaching its minimum near the top of the Δ range, so the calibration rewards high conservatism. At $k = 0.7$ it falls by only about 12%, with a shallow interior minimum around $\Delta = 0.75$. The absolute reductions are in fact similar at the two levels; the difference is that at $k = 0.7$ the reduction is a small fraction of a much larger shortfall, a point developed in Section 6.4.

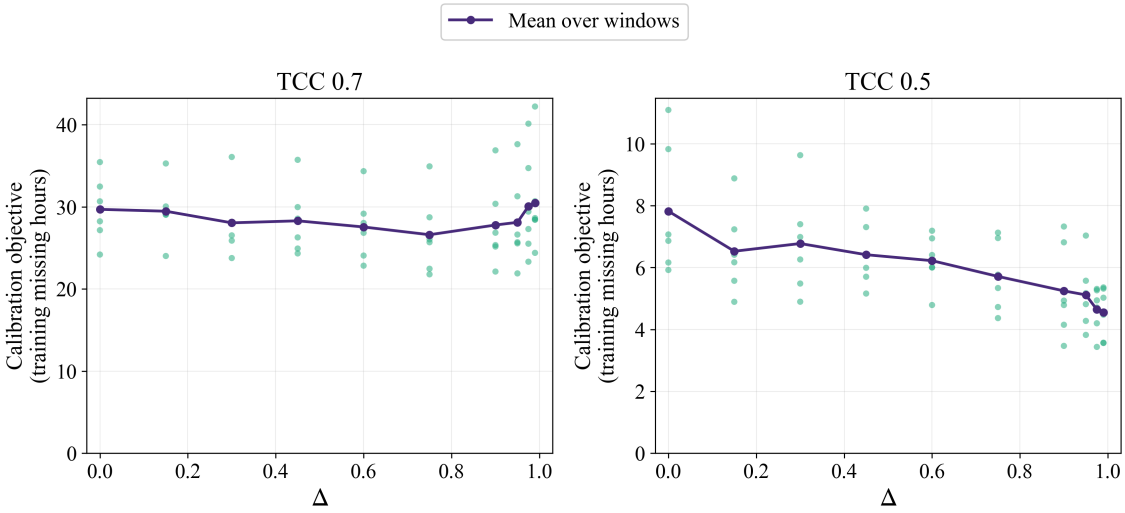

Figure 6.3: Calibration objective (mean missing hours over the training scenarios) against the buffer level Δ, for all six planning windows (dots) and their mean (line), at each constrainedness level.

Search-method diagnostic: grid search versus Bayesian optimisation. Because the global problem is one-dimensional, the exhaustive grid is the natural calibrator: it is inexpensive, covers the tested range completely, and is directly interpretable. It is therefore adopted as the global-buffer specification, independently of the test evaluation. Bayesian optimisation is run on the same case as a diagnostic of the search method (Section 4.6.4): since the grid provides an exhaustive reference in this low-dimensional setting, the comparison shows whether the adaptive search reaches the same optimum, thereby providing a reference point for interpreting the Bayesian results in the block-specific setting.

Table 6.2 compares the two on the calibration objective. At $k = 0.7$ they reach the same value (26.2 hours in both, at the same median $\Delta = 0.75$). At $k = 0.5$ the grid reaches a lower objective (4.2 against 5.2 hours) and the Bayesian search selects a lower median Δ (0.73 against 0.98): it misses the high-conservatism region that the grid reaches by including $\Delta = 0.975$ and $\Delta = 0.99$ explicitly, the near-boundary optimum visible in Figure 6.3. The origin of this shortfall was not

isolated; it is reported as an observation, with possible explanations rather than a demonstrated cause. By construction the buffered requirement is an integer headcount that changes only when a rounded threshold is crossed, so the calibration objective is piecewise-constant in Δ rather than smooth; such a stepped surface, with an optimum near the boundary of the range, is the kind of objective that the smooth surrogate underlying Bayesian optimisation can model poorly. A limited evaluation budget relative to the search space may also contribute. The present experiments do not separate these factors.

This diagnostic carries a caveat into the next section. The block-specific buffer is calibrated by the same Bayesian search, but in five dimensions, where an exhaustive reference is not available. The global case shows that this search can miss optima near the upper end of the range, so the block-specific results should be read with that limitation in mind, particularly where the selected block values approach the boundary.

Table 6.2: Grid search versus Bayesian optimisation for the global buffer, as a search-method diagnostic on the calibration objective (means over the six planning windows; values in training missing hours).

	TCC 0.7		TCC 0.5	
	Grid	BO	Grid	BO
Median selected Δ	0.75	0.75	0.98	0.73
Calibration objective (h)	26.2	26.2	4.2	5.2

6.2.2 Global demand buffer out-of-sample performance

Table 6.3 reports the global buffer, calibrated by grid search, against the baseline on the test scenarios. The buffer reduces missing hours at both levels. At $k = 0.7$ the mean falls from 37.9 to 33.3 hours, an improvement of 11.7%, with all six windows improving. At $k = 0.5$ the mean falls from 9.1 to 4.5 hours, an improvement of 49.1%, with all six windows improving. Tail risk moves in the same direction and is examined separately in Section 6.4.6. The benefit is more than four times larger in the less constrained environment, a contrast examined in Section 6.4.

Table 6.3: Out-of-sample performance of the calibrated global buffer on the test scenarios, averaged over the six windows, by constrainedness level. Absolute baseline values are in Table 6.1; the improvement is the mean of the per-window improvements over the baseline.

TCC	Mean (h)	P95 (h)	Worst (h)	Improvement vs baseline (%)
0.7	33.3	37.0	40.4	11.7
0.5	4.5	5.9	7.4	49.1

6.3 Block-specific demand buffer

The global buffer applies one conservatism level to every period. The block-specific buffer relaxes that restriction: it partitions the planning week into a small number of temporal blocks and assigns a separate conservatism level to each, so the requirement can be reshaped more selectively where demand variability is concentrated. Five blocks are used. Four of them isolate the weekday and hour regions that the exploratory diagnostics of Section 5.3 flagged as carrying atypical demand, and a fifth collects the remaining periods; the concrete partition is defined in Section 5.6 and the buffered requirement it produces in Eq. (4.13). The five levels are calibrated by Bayesian optimisation on the training scenarios, under the protocol of Section 4.6, and the calibrated buffer is then evaluated out of sample on the test scenarios, exactly as for the global buffer.

The calibration inherits the limitation identified for the global buffer in Section 6.2.1: given the exhaustive grid as a reference, the Bayesian search was shown there to miss optima near the upper end of the conservatism range. Here the search operates in five dimensions, where no exhaustive reference is available, so the same limitation applies and cannot be checked directly. The results below are therefore the outcome of a calibration of bounded quality.

Table 6.4 reports the calibrated block-specific buffer on the test scenarios. It improves on the baseline at both levels, by 13.0% at $k = 0.7$ and 44.5% at $k = 0.5$, with every window improving over the baseline. Set against the global buffer, the additional granularity does not yield a consistent gain, though both differences are small in absolute terms: the block buffer is within 0.5 missing hours of the global buffer at the tighter level ($+1.2\%$, 32.8 against 33.3) and within 0.4 at the looser one (-9.5%, 4.9 against 4.5). This comparison, and the reason behind it, is examined in Section 6.4.3.

Table 6.4: Out-of-sample performance of the calibrated block-specific buffer on the test scenarios, averaged over the six windows, by constrainedness level. Baseline and global values are in Tables 6.1 and 6.3. Both improvement columns are means of the per-window improvements; the comparison against the global buffer is discussed in Section 6.4.3, where the per-window variation is shown to matter more than the average.

				Improvement (%)	
TCC	Mean (h)	P95 (h)	Worst (h)	vs baseline	vs global
0.7	32.8	36.5	39.8	13.0	+1.2
0.5	4.9	6.4	7.7	44.5	−9.5

6.4 Cross-condition comparison and discussion

The preceding sections reported the deterministic baseline and the two buffered schedules, the global and the block-specific buffer, evaluated out of sample at the two constrainedness levels. This section brings those results together and interprets them. The aim is not only to identify which schedule performs best, but to explain how a demand buffer acts on a fixed workforce, why its

benefit depends on how tightly that workforce is loaded, and how the two buffer specifications compare. The discussion first sets out the reallocation mechanism, then the effect of constrainedness, and finally the comparison between the global and block-specific buffers, before turning to the robustness of these findings.

6.4.1 How the buffer reallocates the fixed workforce

Since the workforce is fixed, the buffer cannot increase the total amount of coverage available; it can only alter the periods to which that coverage is assigned. The calibrated requirement profile $\widehat{R}$ raises the requirement at a subset of periods, and the MIP, constrained to the same eleven workers, accommodates the increase by withdrawing coverage from elsewhere. Figure 6.4 illustrates this mechanism for a representative window on its busiest weekday, at both levels of constrainedness.

Within each panel the total coverage of the baseline and the buffered schedule is identical (2904 worker-periods), which confirms that the buffer reallocates coverage rather than augmenting it. The volume reallocated is larger at the looser level, amounting to 70 worker-periods at $k = 0.7$ and 106 at $k = 0.5$ on the day shown, reflecting the more aggressive buffer selected there ($\Delta^* = 0.975$ against 0.75). Any benefit from the buffer comes from where that coverage is placed across the day, not from providing more of it. Where the workforce has slack, this reallocation follows the period-level variability profile on which the buffer is constructed: at $k = 0.5$ the coverage that is added falls on periods of markedly higher demand variability than the coverage that is removed (mean period standard deviation 1.31 against 0.84 over the week, with the change in coverage correlating with variability at Pearson correlation $r = 0.38$). At the tighter level this pattern is largely absent ($r = 0.03$; mean period standard deviation 1.22 against 1.16): with the workforce already near saturation there is little idle coverage to redeploy, so the reallocation is constrained by capacity rather than guided by variability, as developed in Section 6.4.2. Coverage is conserved over the planning week rather than within any single day: on the day shown the buffer is close to balanced at $k = 0.7$ but adds more than it removes at $k = 0.5$, the surplus being drawn from lower-demand days such as the Sunday.

The two panels also reveal the structural difference on which the following section builds. At $k = 0.7$ the baseline coverage already follows demand closely throughout the day, leaving little idle coverage available for redeployment; over the window the deployed coverage exceeds mean demand by approximately an eighth, corresponding to a utilisation of 89%. At $k = 0.5$ the same coverage lies well above mean demand at most periods, leaving a pronounced margin, with a utilisation of 68%. It is this margin that governs how much the reallocation achieves, as discussed in Section 6.4.2.

6.4.2 Effect of constrainedness

The benefit of the global buffer depends strongly on constrainedness. At the looser level $k = 0.5$ the calibrated buffer reduces mean missing hours by 49% and improves every window; at the tighter level $k = 0.7$ it reduces them by only 12% (Table 6.3). What distinguishes the two levels is

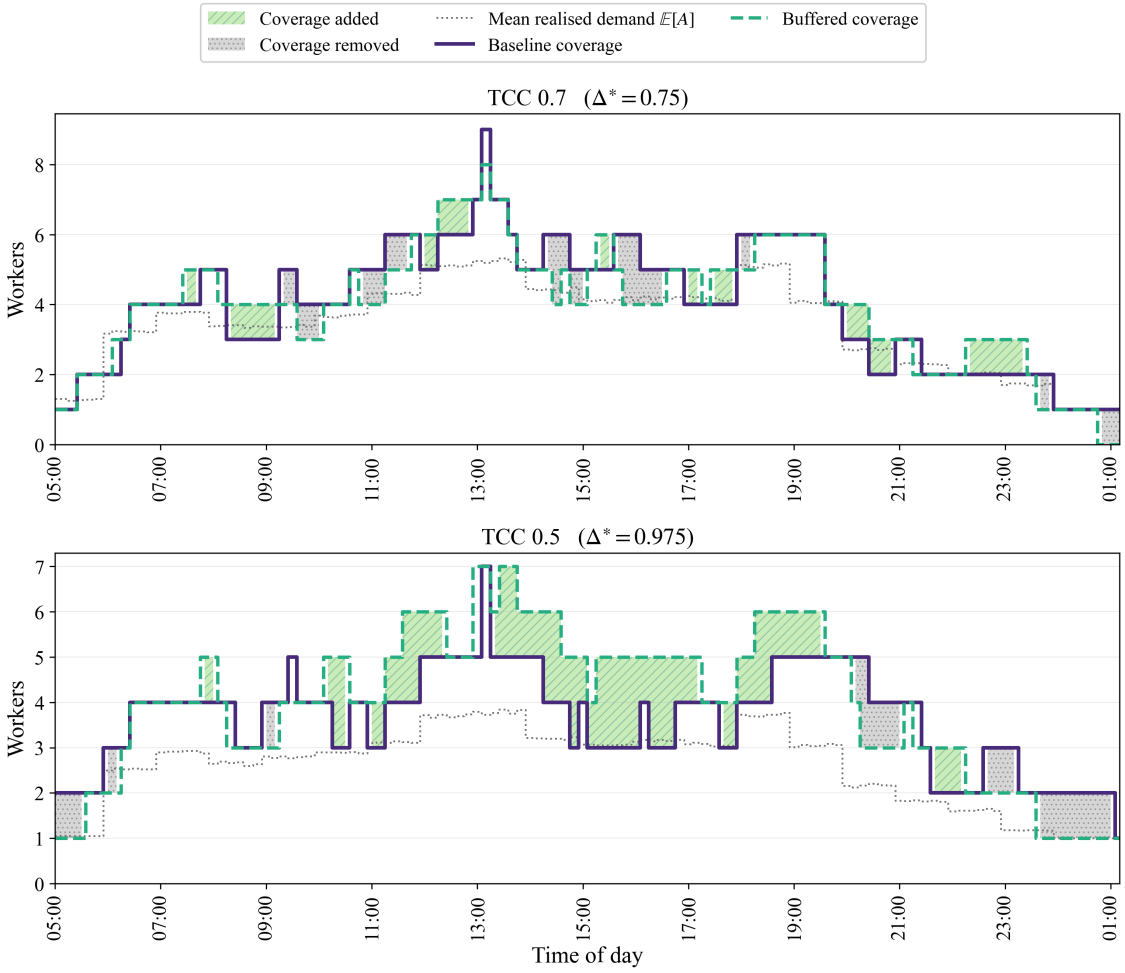

Figure 6.4: Reallocation of the fixed workforce by the global buffer, for a representative window on its busiest weekday (Wednesday), at both constrainedness levels. The solid line denotes the baseline coverage ($\Delta = 0$), the dashed line the buffered coverage at the calibrated Δ^*, and the dotted line the mean realised demand $\mathbb{E}[A]$. Shaded regions mark the periods at which the buffer adds coverage relative to the baseline and those at which it removes coverage. The total coverage is identical for the baseline and buffered schedules within each panel (2904 worker-periods), so the buffer reallocates coverage rather than adding it.

this relative improvement rather than the absolute reduction, which is of comparable size at both (of the order of four to five hours per window): the same absolute reduction is a small fraction of the much larger baseline shortfall at $k = 0.7$ and a large fraction of the small shortfall at $k = 0.5$, the baseline being about four times larger under the tighter level (37.9 against 9.1 missing hours).

The difference is not one of raw capacity. At neither level does mean demand exhaust the workforce, and the missing hours arise from the placement of coverage relative to where demand falls rather than from an absolute shortage of staff. The question is therefore why the buffer recovers so much larger a share of this timing-driven shortfall at $k = 0.5$ than at $k = 0.7$.

The answer is how much spare coverage the schedule has to absorb the reallocation, and how far the buffer's requirement can be raised before it exceeds what the workforce can deliver. Two quantities must be kept apart here. The coverage, that is, what the eleven workers actually staff, is fixed at 2904 worker-periods in every schedule of seven days and is always full. What the buffer changes is the requirement profile $\widehat{R}$, the target it asks the MIP to cover.

Figure 6.5 plots this target, summed over the window and expressed as a fraction of that fixed coverage, against Δ; it is therefore a picture of how much the buffer asks for, not of what is staffed. At $k = 0.7$ the target rises steeply and meets the available coverage near the selected $\Delta = 0.75$, so beyond that point the buffer asks for more than the eleven workers can provide and the extra request cannot be installed. At $k = 0.5$ the target stays well below the available coverage at every Δ in the calibrated grid (at most about 0.9 even at $\Delta = 0.99$), so over that range whatever the buffer asks for can always be staffed. The benefit itself comes from the shape of the reallocation in Figure 6.4, which moves coverage towards the more variable periods; Figure 6.5 shows only how far that reshaping can be pushed before it runs out of workforce, which is what differs between the two levels.

This also explains why the calibration selects a more aggressive buffer at $k = 0.5$. At $k = 0.7$ the selected Δ is bounded by the available coverage: it cannot rise past the point where the requirement exceeds the workforce, which is why it settles at 0.75. At $k = 0.5$ that bound never binds, so Δ is set by the missing-hours objective alone and rises to the top of the grid (median 0.98). A higher selected Δ therefore does not signify more shortfall available to recover, only that the workforce limit does not bind.

A final question is why the looser level, where the reshaping is installed in full, does not close more than 49% of the shortfall. The calibration objective is flat near the top of the grid: $\Delta = 0.98$ is selected, with 0.99 available and not chosen, so further conservatism does not reduce the missing hours. This holds even though the normal approximation has unbounded support, so that the margin keeps growing as $\Delta \to 1$: the additional requirement it produces falls on the high-variability periods that are already covered in almost all scenarios, so the extra coverage there does not reduce the expected shortfall, and the finite workforce caps any further increase in any case. The residual is therefore the floor of the single-Δ family; whether it is irreducible in a wider sense is taken up in Section 6.4.4.

One feature of how the two levels were constructed should be kept in mind when reading this comparison. The two constrainedness levels were obtained by scaling demand against a fixed es-

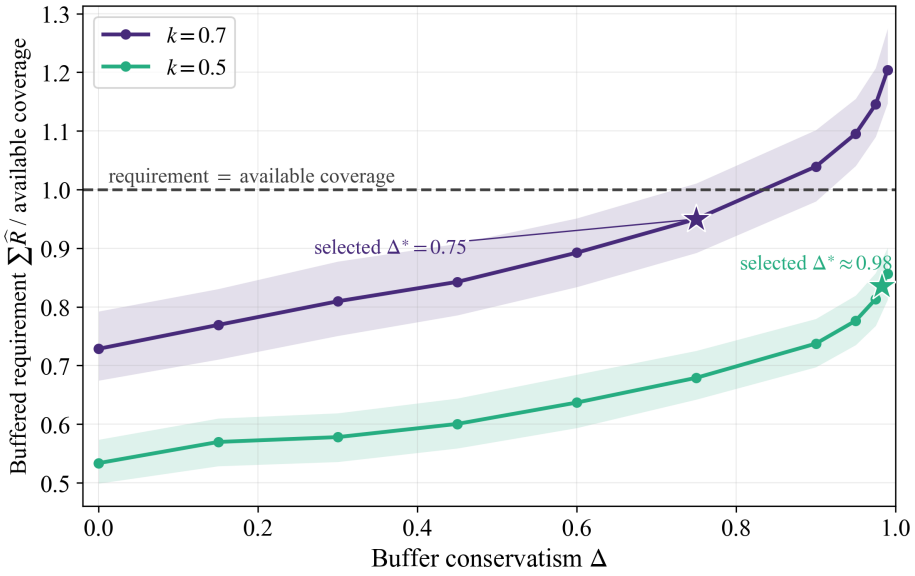

Figure 6.5: How far the buffer's requirement can be raised, by constrainedness level. The curves show the buffered requirement $\sum \widehat{R}$ that the buffer asks the MIP to cover, as a fraction of the fixed coverage the eleven-worker establishment provides (2904 worker-periods), against the conservatism Δ (mean over the six windows; shaded band, window range). The dashed line marks where the requirement equals the available coverage; above it, the requirement exceeds what the workforce can staff. Stars mark the median selected Δ^*. At $k = 0.7$ the requirement meets the available coverage near the selected Δ, bounding how far the buffer can be pushed; at $k = 0.5$ it stays below it at every Δ. The figure concerns the requirement (the target), not the coverage, which is full in every schedule.

tablishment of eleven workers, rather than by varying the size of the workforce at fixed demand, because solving the larger rosters repeatedly across the calibration grid and the test windows was computationally prohibitive. A side effect of scaling is that the two levels differ not only in the demand-to-capacity ratio but also in the absolute magnitude and variability of demand, since scaling the demand scales its standard deviation. This side effect matters for some findings but not for others. The ratio-based findings above, that the buffer's value depends on the spare coverage available and on the point at which the buffered requirement meets the workforce, are unaffected by it, because a schedule is tightened equally by adding demand or by removing workers. Only findings that would instead depend on the absolute level of demand variability are specific to this operationalisation, a point returned to among the limitations in Chapter 7.

6.4.3 Global versus block-specific buffer

The block-specific buffer matches the global buffer at the tighter level and is somewhat worse at the looser one (Table 6.4). The added temporal granularity therefore does not pay off on these data, and a likely reason is that the global buffer is already period-specific. Although it applies a single scalar Δ, the requirement it produces is not raised uniformly: the increase at each period is proportional to that period's demand standard deviation (Eq. (4.12)), so the buffer already concentrates the reshaping on the more variable periods, which is what the reallocation of Section 6.4.2 shows. The partition adds freedom only by letting the conservatism differ between groups of periods, on top of the period-level shaping the global buffer already performs. With most of the available reshaping captured by the variance-proportional global buffer, the partition has little left to exploit.

The per-window outcomes vary more than the averages suggest, and the variation is informative. At $k = 0.7$ the block buffer outperforms the global buffer in two windows by a clear margin (windows 1 and 5, by 2.8 and 1.6 missing hours), is level in two, and is slightly behind in two; at $k = 0.5$ it is behind in five of six windows. The two clear gains are not calibration noise: in both windows the block buffer improves on the global one in-sample as well as out of sample, the signature of a solution that generalises rather than one that overfits the training scenarios.

This pattern admits a precise reading in-sample, because the block buffer contains the global buffer as a special case: setting all five conservatism levels equal reproduces the single-Δ buffer. A perfectly calibrated block buffer could therefore never be worse than the global buffer on the calibration objective, since it could always fall back to the uniform setting. Where the block buffer is in fact worse in-sample, which occurs in seven of the twelve window-level cases, the cause is necessarily the calibration: the five-dimensional search did not reach the uniform-equivalent point. The in-sample shortfall is thus a limitation of the search, not of the partition.

Out of sample the attribution is not as clean, and the present design cannot settle it. One is sampling variance: with five free parameters rather than one, the calibration can fit the particular training scenarios more closely, so an in-sample advantage need not carry to the test set. This is variance from the extra degrees of freedom, not fitting to any individual block, since the calibration optimises a single training-average objective over all periods. The other is structural: the partition

is held fixed across windows, while the deterministic requirement each window is built on is window-specific, so one fixed partition can align with a given window's demand profile better than with another's. A single test set without replication does not separate them. The block buffer therefore shows a real but unreliable advantage, whose source, calibration or partition, cannot be isolated here.

Where the block buffer does win, the margin is not an artefact of unconverged solves. In window 1 at $k = 0.7$ the block improvement over the global buffer (2.8 missing hours) exceeds the global solve's entire unproven optimality room (about 2.0 hours; Appendix D), so even crediting the global solve with the full benefit of a longer run would not overturn the block buffer's advantage there. The block solves are themselves near-optimal in absolute terms (Appendix D), so the comparison reflects the buffer specification rather than the solver. Whether a better-calibrated or differently chosen partition would do better remains open; how much room any buffer leaves above it is taken up next.

6.4.4 How much room remains: a non-anticipative benchmark

To judge whether the calibrated buffer is close to the best a fixed schedule can do, it is compared against the non-anticipative benchmark introduced in Section 4.7, which optimises a single fixed roster directly on the training demand distribution, with no requirement profile and no buffer (Appendix E). The benchmark is non-anticipative in the same sense as the buffered schedules, committing to one roster before demand is realised, but it uses different information: it is fitted to historical demand rather than to the planned requirement the buffer reshapes. It is therefore a diagnostic, not a competing method, and it answers one question: does room remain above the buffer?

The answer is yes, at both levels. The benchmark attains less missing coverage than either buffer in every window (Appendix E). Averaged over windows, it reduces the baseline shortfall by 29.8% at $k = 0.7$ and 73.1% at $k = 0.5$, against 11.7% and 49.1% for the global buffer; in absolute terms it reaches 26.6 against 33.3 missing hours at $k = 0.7$ and 2.4 against 4.5 at $k = 0.5$. The buffer is thus far from the best non-anticipative schedule the same data permit.

This has two consequences. First, it revises the reading of the residual at $k = 0.5$ in Section 6.4.2. That residual is not the price of non-anticipativity: the benchmark is itself non-anticipative and reaches well below the buffer, so the gap is the distance between the buffer-induced roster and the distribution-optimal fixed roster, not an irreducible floor that only recourse could remove. Second, the comparison is one-sided. Had the benchmark been close to the buffer, it would have shown that no buffer refinement, finer blocks included, could help. Because it is not, room demonstrably exists; but the benchmark is an unconstrained roster, not a buffer, so it does not show that any buffer, block or otherwise, can reach it. Whether the open room is accessible to a better-calibrated or finer buffer is left unresolved.

A final caveat concerns what the benchmark optimises. It is fitted to realised historical demand and discards the planned requirement, whereas the buffer keeps that requirement and only reshapes it, so the room reported here is room under historical information. The same distinction

bears on how the two approaches would compare in deployment. The benchmark replaces the planned requirement with a stochastic optimisation against the historical record, discarding whatever forward-looking information the requirement carries; the buffer robustifies that requirement instead of replacing it, and so retains it. Where realised demand reflects upcoming conditions that the planned requirement anticipates and the historical record does not, this retention is a reason to robustify an existing forecast rather than replace it with a purely historical optimiser. The present evaluation, run against held-out historical demand, can neither exercise nor measure that advantage; its implications are drawn together in Chapter 7.

6.4.5 Robustness: solver gaps and test difficulty

Most schedules were obtained under a time limit without a proven optimum (Appendix D), so it is worth confirming that the comparisons are not solver artefacts. The two buffer specifications are calibrated under different per-solve time limits, 7200 seconds for the grid and global searches and 900 seconds for the block-specific search (Section 5.7). This difference does not undermine the comparison, because the solver reaches its final schedule well before either limit. Across the 7200-second grid and global solves, the incumbent is reached within a median of 186 seconds, and within roughly 780 seconds in 95 percent of cases; the remaining time is spent proving optimality rather than improving the schedule. The 900-second limit therefore captures the incumbent in 96 percent of the profiled solves (Table D.3), so a solve run under it yields essentially the schedule it would have reached under the longer limit. Any difference between the global and block-specific buffers can thus be attributed to the buffer specification rather than to the time budget under which each was calibrated.

The absolute optimality room is small in every window for all three schedules, baseline, global and block (at most about 2.2 hours; Tables D.1 and D.2), and a per-window reading shows that the unproven gaps never reverse a conclusion. This holds despite the block-specific solves running under the shorter time limit: their absolute room is at most 1.8 hours at $k = 0.7$ and below 0.4 hours at $k = 0.5$, where the larger percentage gaps rest on near-zero objectives and therefore do not signal a poorly placed roster. Where the global buffer helps least ($k = 0.7$, windows 1 and 5), its grid-selected solve is in fact the least well solved of the window; since a tighter solve can only lower its missing, the reported improvement is a lower bound rather than an over-statement, and in window 1 the block buffer's advantage over the global one exceeds that room (Section 6.4.3). Finally, the test scenarios are not easier than the calibration scenarios (Appendix A), so the out-of-sample improvements are not flattered by an undemanding test period.

6.4.6 Average versus tail risk

Both buffers lower tail risk together with the average, and neither does so by trading one against the other. At both constrainedness levels the 95th percentile and the worst case fall in the same direction as the mean, and by a similar proportion, with the reduction slightly smaller at the tail and smallest at the worst case (Tables 6.3 and 6.4). For the global buffer the mean, the 95th percentile

and the worst case improve by about 12%, 11% and 10% at $k = 0.7$ and by about 49%, 44% and 40% at $k = 0.5$; for the block buffer the corresponding figures are about 13%, 12% and 11% at $k = 0.7$ and 45%, 40% and 38% at $k = 0.5$. The two specifications behave alike, and in both the tail improves almost as much as the average.

This ordering is consistent with the shape of the reallocation rather than from any safety margin: the buffer trims missing coverage across the whole distribution by shifting coverage to the variable periods, but the most extreme realisations exceed coverage by margins that a fixed reallocation cannot fully offset, so they improve by a slightly smaller proportion than the bulk of the distribution.

The absence of an average-versus-tail trade-off is itself informative. A buffer is often understood as a costly safety margin that protects against extreme outcomes at the expense of average performance (Ingels and Maenhout, 2019; De Bruecker et al., 2015; Bertsimas and Sim, 2004). Here it is not such a margin. Because the workforce is fixed, the buffer does not add coverage; it reallocates a constant coverage budget towards the more variable periods, which are precisely the periods that generate the high-shortfall scenarios driving the tail. The central and the extreme scenarios therefore improve at the same time and at no cost in expected coverage. The improvement reported in this chapter is thus a direct consequence of placing the same workforce where variability is highest, rather than a risk premium paid for tail protection.

Taken together, the results of this chapter give a consistent picture. On a fixed workforce, the demand buffer reduces out-of-sample shortfall at both constrainedness levels without adding any coverage, by reshaping a constant coverage budget towards the periods where realised demand is most variable. The size of this gain depends on how tightly the workforce is loaded: it is large when the workforce has slack (49.1% at $k = 0.5$) and modest when it is tightly loaded (11.7% at $k = 0.7$), and it lowers the tail of the shortfall distribution by almost as much as the mean, with no trade-off between the two. Increasing the temporal granularity through a block-specific partition does not appear to improve on the variance-proportional global buffer, which already concentrates the reshaping on the variable periods. A non-anticipative diagnostic indicates that room remains above the buffer under historical information; because it is fitted to the realised history rather than to the planning forecast the buffer retains, it shows that room exists without establishing that any buffer can reach it. On a fixed workforce, then, the buffer reshapes an existing forecast at no capacity cost, most clearly when the workforce has slack.

Chapter 7

Conclusions and Future Work

This dissertation asked whether demand-buffer robustification of a deterministic staff-scheduling model can improve its performance under stochastic demand without changing the model, and under what conditions. This chapter states the conclusions that answer that question, the limitations of the study, and the work they motivate. Each conclusion follows from the results and discussion of Chapter 6.

7.1 Conclusions

The demand buffer improves the out-of-sample stochastic performance of the deterministic schedule at both constrainedness levels, without altering the model or adding capacity. It reduces expected shortfall by 11.7% under tight loading and 49.1% under slack loading, and it lowers the 95th percentile and the worst case by almost as much as the mean, so the improvement carries no average-versus-tail trade-off (Sections 6.2.2 and 6.4.6).

This improvement comes from reallocation, not addition. Because the workforce is fixed, the buffer cannot increase total coverage; it redistributes a constant coverage budget, shifting it towards the more variable periods where the workforce has slack to do so (Section 6.4).

The size of the benefit depends on how tightly the workforce is loaded. The buffer recovers a large share of the shortfall when the workforce has slack and only a modest share when it is tightly loaded, because the spare coverage available sets how far the requirement can be reshaped before it exceeds what the workforce can staff (Section 6.4.2).

Finer temporal granularity did not yield a dependable advantage on these data. A block-specific buffer, calibrated by Bayesian optimisation over five temporal blocks, matched the global buffer at the higher constrainedness level and was slightly worse at the lower one, with genuine gains in a few windows but no reliable improvement on average. A likely reason is that the global buffer is already period-specific: it raises each period's requirement in proportion to that period's demand variability, leaving the partition little to exploit. Whether the source is this structural redundancy or the bounded quality of the five-dimensional calibration cannot be separated here (Section 6.4.3).

A non-anticipative diagnostic indicates that room remains above the buffer under historical information. Fitted to the realised demand distribution rather than to the planning forecast the buffer retains, this reference attains less shortfall than either buffer, but, optimising under different information and replacing the requirement rather than reshaping it, it establishes that room exists without showing that any buffer can reach it (Section 6.4.4).

Taken together, demand-buffer robustification is an effective and low-cost way to make a deterministic scheduler more robust to stochastic demand. It requires no change to the optimisation model, is most valuable when the workforce is not tightly loaded, does not benefit from finer temporal granularity on these data, and leaves room, under historical information, that a richer scheme might in principle recover.

In practice, the method is the calibration procedure rather than a fixed buffer value. A post estimates its period-level demand variability from historical data and calibrates the conservatism level Δ against simulated stochastic performance, producing one buffer for that post; the calibrated value is specific to the instance and is not read from its constrainedness, since the selected Δ varies across windows at the same TCC (Appendix C). TCC instead serves as a diagnostic of the expected benefit, large when the workforce has slack and modest when it is tightly loaded, which lets the value of applying the buffer be anticipated before it is calibrated. Because the global buffer is a single scalar, this calibration requires only a small number of deterministic solves, far cheaper than building a stochastic reformulation.

7.2 Limitations

The study has five main limitations, each developed in Chapter 6.

First, the two constrainedness levels were created by scaling demand against a fixed workforce rather than by varying the workforce at fixed demand, for reasons of computational cost (Section 6.4.2). The ratio-based findings are unaffected, but any finding that depends on the absolute level of demand variability is specific to this operationalisation.

Second, the block-specific buffer was calibrated by Bayesian optimisation in five dimensions, where no exhaustive reference exists and the search was shown to miss the optimum even in the one-dimensional case (Section 6.2.1). Part of the block buffer's underperformance is therefore attributable to calibration rather than to the partition (Section 6.4.3).

Third, the block buffer's temporal partition was fixed in advance and held constant across windows. Whether a finer or window-adaptive partition would capture the room the benchmark reveals is not resolved (Section 6.4.3).

Fourth, schedules were evaluated against held-out historical demand. This cannot credit the forward-looking information that the planned requirement may carry, which is a central reason to robustify an existing forecast rather than replace it with a purely historical optimiser (Section 6.4.4).

Fifth, the buffer can only raise the requirement, never lower it, so it cannot correct periods where the planned requirement overstates demand (Section 4.5.3).

7.3 Future work

These limitations point to several directions. Firstly, a complementary design that varies the workforce at fixed demand would separate the capacity effect from the demand-variability effect that the present operationalisation combines. A stronger calibration, whether a better-converged search or one tailored to the step-like objective, together with finer or window-adaptive partitions, would test whether the block buffer's isolated gains can be made dependable. Allowing the buffer to lower as well as raise the requirement would let it correct an over-stated forecast, not only an understated one. Introducing per-scenario recourse would move beyond the non-anticipative ceiling that bounds any fixed roster, the buffer included. Finally, evaluating the buffer against genuine future demand, rather than held-out historical demand, would assess the forward-looking value that the present evaluation cannot measure and that motivates robustifying a forecast rather than replacing it.

More broadly, these directions place demand-buffer robustification at the intersection of several active research lines. The study addresses demand uncertainty alone. Therefore, extending the robustification to capacity and availability uncertainty, most notably worker absenteeism (Green et al., 2013; Ingels and Maenhout, 2019), whether through capacity reserves or a combined demand-and-availability buffer, is a natural next step, and distributionally robust formulations have recently been applied to staffing precisely under absenteeism (Ryu and Jiang, 2025). Besides that, the static buffer calibrated here could become a contextual, learned policy, $\Delta = f(x)$, that sets the buffer from operational covariates such as weekday, hour or the constrainedness of the instance, connecting the approach to prescriptive analytics and decision-focused learning (Bertsimas and Kallus, 2019; Elmachtoub and Grigas, 2021; Ban and Rudin, 2019). Under this view TCC would act not only as a diagnostic of the expected benefit but as a feature driving the buffer itself. The simulation-calibrated buffer could also be benchmarked against distributionally robust formulations, which protect against a set of plausible demand distributions rather than a single empirical one, giving a formal counterpart to the pragmatic enhancement studied here (Delage and Ye, 2010; Mohajerin Esfahani and Kuhn, 2018). Together these situate the buffer as a low-cost, deployable point on a spectrum that runs from pragmatic input reshaping to fully data-driven and formally robust staffing.

Bibliography

Abernathy, W. J., Baloff, N., Hershey, J. C., and Wandel, S. (1973). A three-stage manpower planning and scheduling model—a service-sector example. *Operations Research*, 21(3):693–711.

Acquaye, A. (2025). Operational research for sustainability: a synthesis of methods, applications and challenges. *Journal of the Operational Research Society*, 77(1).

Ağralı, S., Taşkın, Z. C., and Ünal, A. T. (2017). Employee scheduling in service industries with flexible employee availability and demand. *Omega*, 66:159–169.

Amaran, S., Sahinidis, N. V., Sharda, B., and Bury, S. J. (2016). Simulation optimization: a review of algorithms and applications. *Annals of Operations Research*, 240(1):351–380.

Avramidis, A. N., Chan, W., Gendreau, M., LÉcuyer, P., and Pisacane, O. (2010). Optimizing daily agent scheduling in a multiskill call center. *European journal of operational research*, 200(3):822–832.

Ban, G.-Y. and Rudin, C. (2019). The big data newsvendor: Practical insights from machine learning. *Operations Research*, 67(1):90–108.

Bertsimas, D. and Kallus, N. (2019). From predictive to prescriptive analytics. *Management Science*, 66(3):1025–1044.

Bertsimas, D. and Sim, M. (2004). The price of robustness. *Operations Research*, 52(1):35–53.

Billionnet, A. (1999). Integer programming to schedule a hierarchical workforce with variable demands. *European journal of operational research*, 114(1):105–114.

Chen, W., Gao, S., Chen, W., and Du, J. (2023). Optimizing resource allocation in service systems via simulation: A bayesian formulation. *Production and Operations Management*, 32(1):65–81.

De Bruecker, P., Van den Bergh, J., Beliën, J., and Demeulemeester, E. (2015). A model enhancement heuristic for building robust aircraft maintenance personnel rosters with stochastic constraints. *European Journal of Operational Research*, 246(2):661–673.

Delage, E. and Ye, Y. (2010). Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612.

Direção-Geral de Economia (2026). Síntese estatística setorial: Cae 471 – comércio a retalho em estabelecimentos não especializados. Acesso em 2026-06-25.

Doneda, M., Smet, P., Carello, G., Lanzarone, E., and Vanden Berghe, G. (2026). Robust personnel rostering: how accurate should absenteeism predictions be? *Journal of Scheduling*, 29(1):67–82.

Duffuaa, S. O. and Al-Sultan, K. (1999). A stochastic programming model for scheduling maintenance personnel. *Applied Mathematical Modelling*, 23(5):385–397.

Easton, F. F. (2014). Service completion estimates for cross-trained workforce schedules under uncertain attendance and demand. *Production and Operations Management*, 23(4):660–675.

Elmachtoub, A. N. and Grigas, P. (2021). Smart "predict, then optimize". *Management Science*, 68(1):9–26.

Ernst, A., Jiang, H., Krishnamoorthy, M., and Sier, D. (2004). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*, 153(1):3–27. Timetabling and Rostering.

Figueira, G. and Almada-Lobo, B. (2014). Hybrid simulation–optimization methods: A taxonomy and discussion. *Simulation modelling practice and theory*, 46:118–134.

Frantzén, M., Ng, A. H., and Moore, P. (2011). A simulation-based scheduling system for real-time optimization and decision making support. *Robotics and Computer-Integrated Manufacturing*, 27(4):696–705.

Green, L. V., Savin, S., and Savva, N. (2013). Nursevendor problem: Personnel staffing in the presence of endogenous absenteeism. *Management science*, 59(10):2237–2256.

Ingels, J. and Maenhout, B. (2015). The impact of reserve duties on the robustness of a personnel shift roster: An empirical investigation. *Computers & Operations Research*, 61:153–169.

Ingels, J. and Maenhout, B. (2017). Employee substitutability as a tool to improve the robustness in personnel scheduling. *OR Spectrum*, 39(3):623–658.

Ingels, J. and Maenhout, B. (2018). The impact of overtime as a time-based proactive scheduling and reactive allocation strategy on the robustness of a personnel shift roster. *Journal of Scheduling*, 21(2):143–165.

Ingels, J. and Maenhout, B. (2019). Optimised buffer allocation to construct stable personnel shift rosters. *Omega*, 82:102–117.

Jongbloed, G. and Koole, G. (2001). Managing uncertainty in call centres using poisson mixtures. *Applied Stochastic Models in Business and Industry*, 17(4):307–318.

Liao, S., van Delft, C., and Vial, J.-P. (2013). Distributionally robust workforce scheduling in call centres with uncertain arrival rates. *Optimization Methods and Software*, 28(3):501–522.

Maenhout, B. and Vanhoucke, M. (2013). An integrated nurse staffing and scheduling analysis for longer-term nursing staff allocation problems. *Omega*, 41(2):485–499.

Maheut, J., Garcia-Sabater, J. P., Garcia-Sabater, J. J., and Garcia-Manglano, S. (2024). Solving the multisite staff planning and scheduling problem in a sheltered employment centre that employs workers with intellectual disabilities by milp: a spanish case study. *Central European Journal of Operations Research*, 32(3):569–591.

Mani, V., Kesavan, S., and Swaminathan, J. M. (2015). Estimating the impact of understaffing on sales and profitability in retail stores. *Production and Operations Management*, 24(2):201–218. Acesso em 2026-06-25.

Manteu, C., Antunes, A., Castro, G., Félix, S., et al. (2021). A estrutura de custos operacionais das empresas portuguesas. *Revista de Estudos Económicos*. Banco de Portugal; acesso em 2026-06-25.

Mattia, S., Rossi, F., Servilio, M., and Smriglio, S. (2017). Staffing and scheduling flexible call centers by two-stage robust optimization. *Omega*, 72:25–37.

Miranda, J., Rey, P. A., Sauré, A., and Weber, R. (2018). Metro uses a simulation-optimization approach to improve fare-collection shift scheduling. *Interfaces*, 48(6):529–542.

Mohajerin Esfahani, P. and Kuhn, D. (2018). Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171:115–166.

OECD (2026). Oecd economic surveys: Portugal 2026 – basic statistics of portugal, 2024. Acesso em 2026-06-25.

Parisio, A. and Jones, C. N. (2015). A two-stage stochastic programming approach to employee scheduling in retail outlets with uncertain demand. *Omega*, 53:97–103.

Powell, W. B. (2022). Cost function approximations. In *Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions*, chapter 13. John Wiley & Sons, Hoboken, NJ.

Powell, W. B. and Ghadimi, S. (2022). The parametric cost function approximation: A new approach for multistage stochastic programming. *arXiv preprint arXiv:2201.00258*.

Ramanujam, A. and Li, C. (2025). Pamso: Parametric autotuning multi-time scale optimization algorithm. *Computers & Chemical Engineering*, 200:109160.

Restrepo, M. I., Gendron, B., and Rousseau, L.-M. (2017). A two-stage stochastic programming approach for multi-activity tour scheduling. *European Journal of Operational Research*, 262(2):620–635.

Robbins, T. R. and Harrison, T. P. (2010). A stochastic programming model for scheduling call centers with global service level agreements. *European Journal of Operational Research*, 207(3):1608–1619.

Ryu, M. and Jiang, R. (2025). Nurse staffing under absenteeism: A distributionally robust optimization approach. *Manufacturing & Service Operations Management*, 27(2):624–639.

United Nations (2015). Transforming our world: The 2030 agenda for sustainable development. General Assembly resolution A/RES/70/1.

Van den Bergh, J., Beliën, J., De Bruecker, P., Demeulemeester, E., and De Boeck, L. (2013). Personnel scheduling: A literature review. *European journal of operational research*, 226(3):367–385.

Warner, D. M. and Prawda, J. (1972). A mathematical programming model for scheduling nursing personnel in a hospital. *Management Science*, 19(4-part-1):411–422.

World Bank (2026). Employment in services (% of total employment) (modeled ilo estimate) – indicator sl.srv.empl.zs. Fonte de base: ILOSTAT; acesso em 2026-06-25.

Sustainable Development Goals

The contribution of operational research to sustainable development is widely acknowledged, yet broadening its engagement with that agenda remains an open challenge for the field (Acquaye, 2025). This dissertation does not set out to address sustainability directly; its immediate aim is to improve a scheduling decision. The decisions it improves, however, bear on working conditions, on the efficient use of labour, and on the resilience of an organisation's existing planning tools, and personnel scheduling is itself concerned with the workload and well-being of staff, not only with service coverage (Ernst et al., 2004; Van den Bergh et al., 2013). The 2030 Agenda for Sustainable Development and its seventeen Sustainable Development Goals (SDGs) give a common reference for concerns of this kind (United Nations, 2015). The goals to which the work is most relevant, the targets within them, and the metrics through which the contribution can be read, are set out below.

Table 7.1: Contribution of this dissertation to the Sustainable Development Goals

SDG	**Target**	**Contribution**	**Performance indicators and metrics**
SDG 8: Decent Work and Economic Growth	8.2 Higher economic productivity through technological upgrading and innovation	A fixed workforce covers more of the realised demand through a low-cost robustification of the deterministic scheduler, raising labour productivity without adding headcount.	Out-of-sample expected missing hours and improvement over the deterministic baseline (11.7% under tight loading, 49.1% under slack loading); service level.
	8.8 Safe and secure working environments and labour rights	Schedules are produced within the contractual working-time, rest and break rules, and lower understaffing reduces the overload borne by the staff on shift.	Adherence to the model's labour rules; reduction in the worst-case (peak) missing hours.
SDG 9: Industry, Innovation and Infrastructure	9.5 Enhance research and upgrade technological capabilities	A simulation-based calibration and robustification layer strengthens an existing deterministic MIP without redesigning it.	Improvement over the deterministic baseline; remaining gap to a non-anticipative benchmark.
SDG 3: Good Health and Well-being	3.4 Promote mental health and well-being	Indirectly, lower average and worst-case shortfall may reduce the overtime and pressure that sustained understaffing places on workers.	Reduction in average and worst-case missing hours (proxy; well-being not measured directly).

The common thread across these goals is that the work makes an existing planning process do more with the workforce it already has, within the rules that protect that workforce. Its clearest contribution is to decent work (SDG 8); the links to innovation (SDG 9) and, more indirectly, to well-being (SDG 3) follow from the same mechanism.

Appendix A

Relative difficulty of the training and test sets

The temporal split assigns the first half-year to testing and the second half-year to calibration (training). Because the split is temporal rather than random, the two sets need not be equally demanding. This appendix checks that the test set is not easier than the training set, so that the out-of-sample results of Chapter 6 are not flattered by an undemanding test period. Two indicators are used: the overall level of demand and the frequency of extreme observations.

Averaged over all weekday and hour buckets, mean realised demand is 4.23 workers per period in the test set against 3.97 in the training set. The test period therefore carries a slightly higher central demand.

Extreme observations are identified with the Tukey upper fence: within each weekday and hour bucket, an observation is an outlier if it exceeds $Q_3 + 1.5\,(Q_3 - Q_1)$. The test set contains 1095 such observations against 757 in the training set, even though the two sets span comparable periods (roughly six months each). Figures A.1 to A.3 show these counts resolved by weekday and time of day.

Table A.1: Demand level and frequency of extreme observations in the two sets.

	Training (second half)	Test (first half)
Mean demand per period	3.97	4.23
Upper-fence outliers (count)	757	1095

Both indicators point the same way: the test set has a higher demand level and more frequent extremes than the training set. The out-of-sample evaluation in Chapter 6 is therefore conservative, since configurations calibrated on the training set are assessed on a period that is, if anything, harder. Any improvement that survives this evaluation is correspondingly stronger evidence that the buffers help under demand that the calibration never saw.

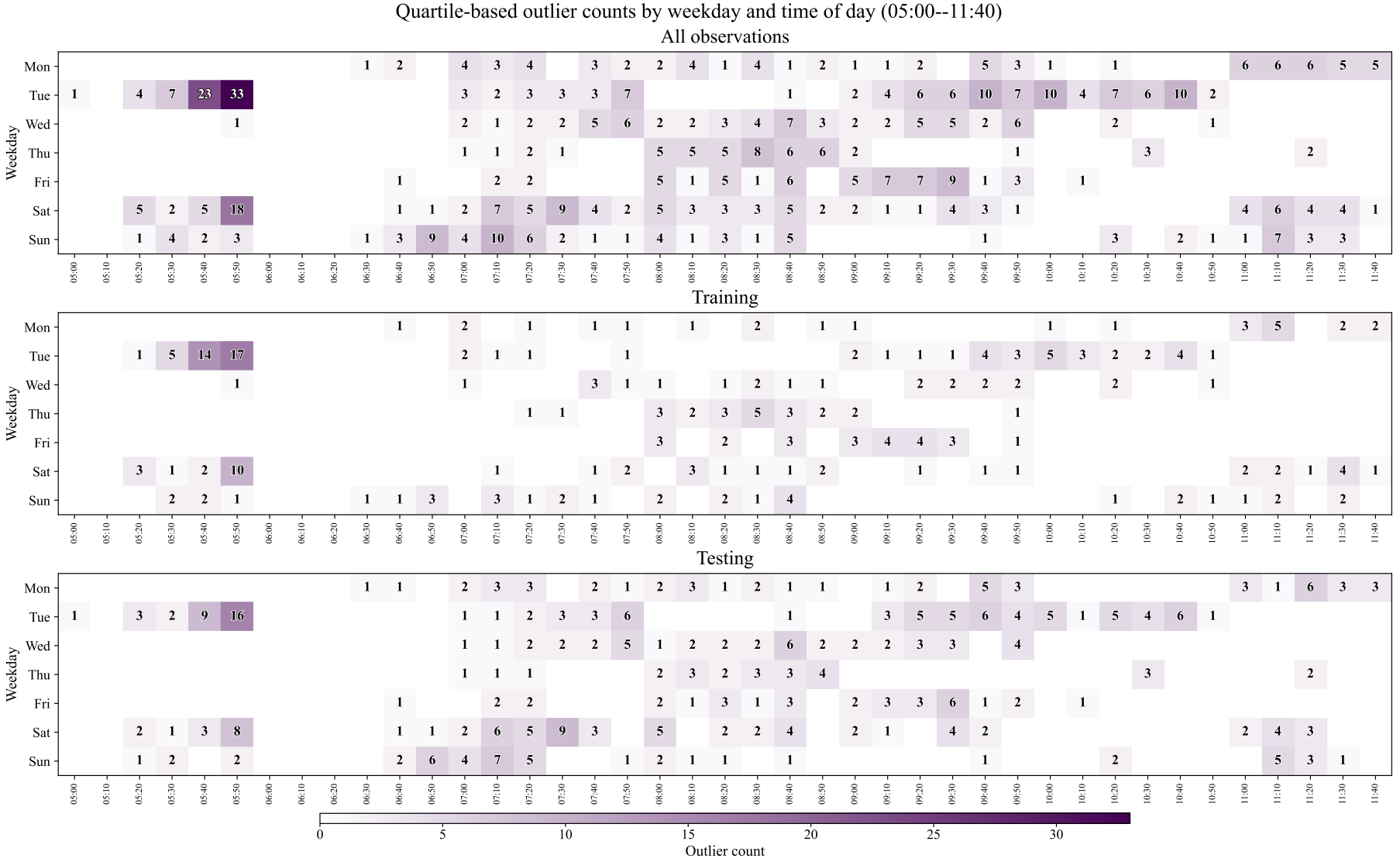

Figure A.1: Quartile-based outlier counts by weekday and time of day, from 05:00 to 11:40. The three panels report all observations, the training subset and the testing subset.

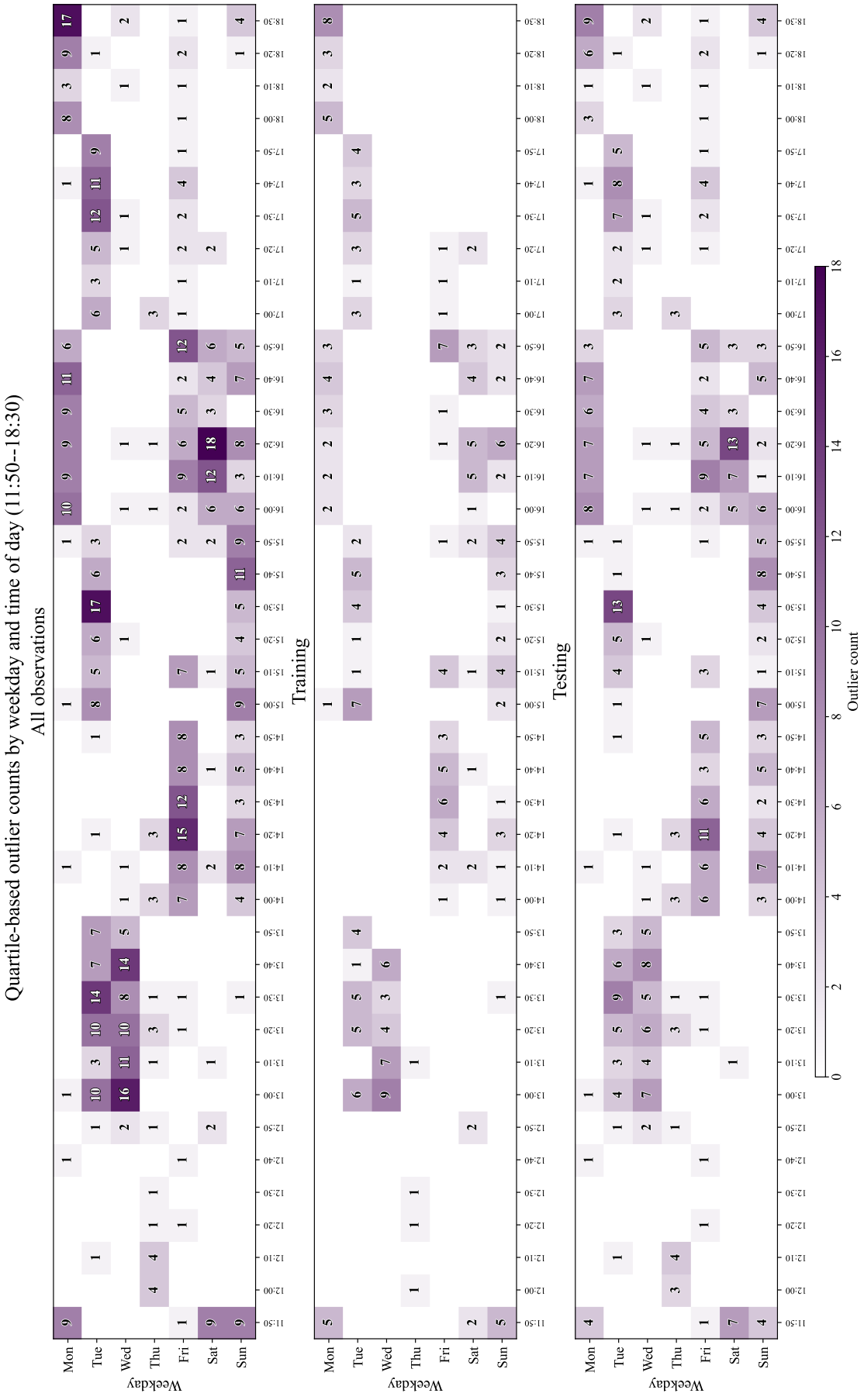

Figure A.2: Quartile-based outlier counts by weekday and time of day, from 11:50 to 18:30. The three panels report all observations, the training subset and the testing subset.

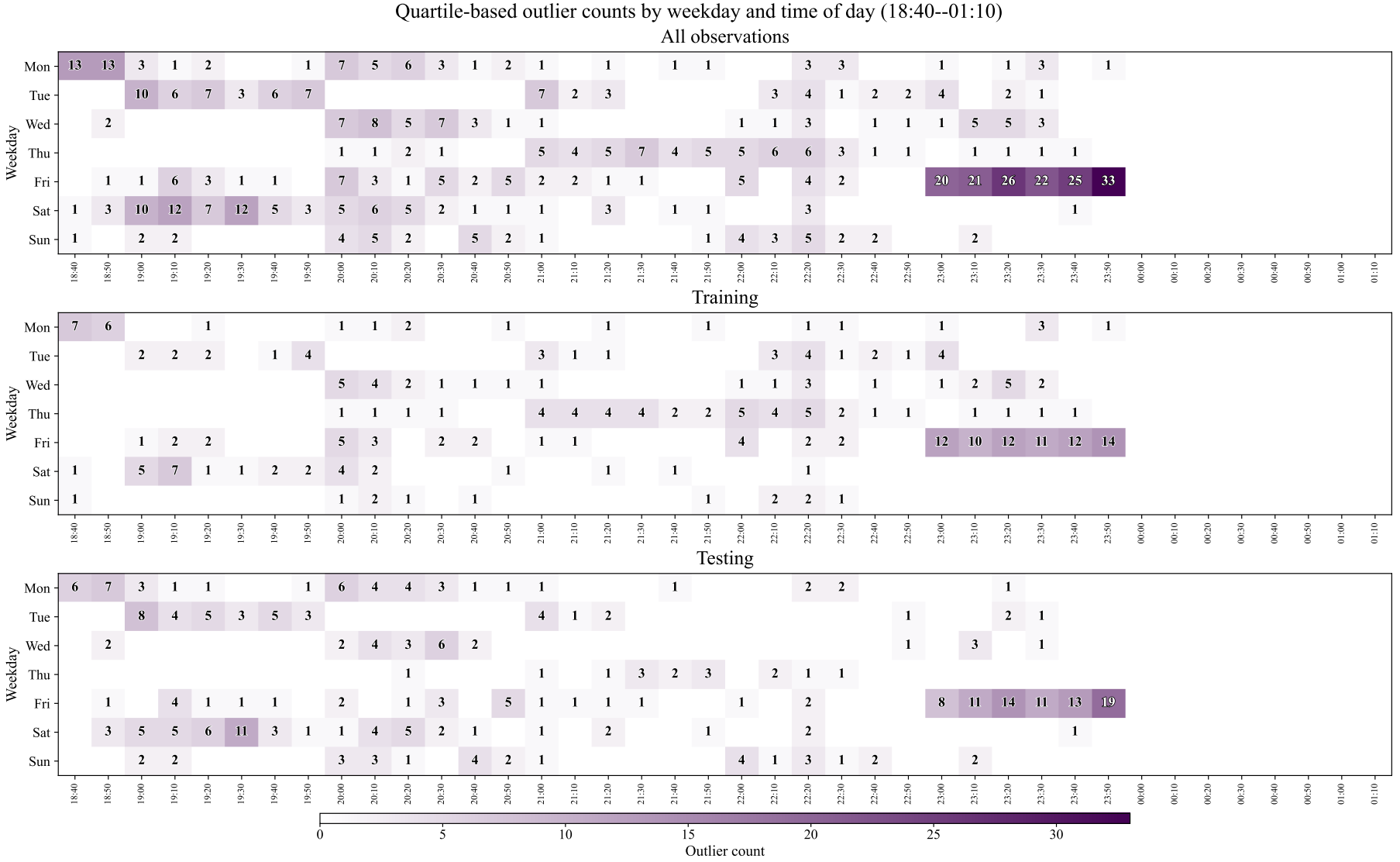

Figure A.3: Quartile-based outlier counts by weekday and time of day, from 18:40 to 01:10. The three panels report all observations, the training subset and the testing subset.

Appendix B

Distributional Approximation Diagnostics

This appendix reports representative graphical diagnostics used to support the distributional approximation adopted for buffer construction in Chapter 5. The objective is not to identify a statistically optimal probability distribution for realised demand, nor to define the stochastic scenario-generation procedure. Instead, the plots provide exploratory visual evidence for the pragmatic use of a normal-based approximation when constructing conservative demand-buffer margins.

For selected weekday–hour combinations, the empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations. The selected figures are representative examples rather than an exhaustive reproduction of all weekday–hour diagnostics. They were chosen to illustrate different demand conditions, including lower-demand periods, morning periods and higher-demand midday periods.

The examples show that no single distribution is uniformly dominant across all inspected buckets. In some cases, such as Friday 06:00 (Figure B.1) and Monday 13:00 (Figure B.3), the Normal and Weibull approximations follow the central mass of the empirical distribution reasonably closely, while Poisson also captures part of the discrete shape of the data. In other cases, such as Monday 08:00 (Figure B.2) and Tuesday 07:00 (Figure B.8), the empirical distribution is more concentrated around a small number of demand levels, and the candidate distributions differ more visibly in the tails. The late-evening examples, such as Saturday 23:00 (Figure B.5) and Sunday 23:00 (Figure B.6), show more strongly skewed low-demand profiles, where the Normal approximation is less visually exact but still provides a simple representation of local variability around the nominal requirement.

Taken together, Figures B.1–B.8 support the interpretation of the Normal approximation as a pragmatic margin-construction device rather than as a complete probabilistic model of demand. The Normal approximation is not visually perfect in every inspected bucket, but the alternative Poisson and Weibull approximations do not appear consistently superior across the selected examples. For this reason, the Normal approximation was retained as a simple, interpretable and sufficiently reasonable basis for constructing demand-buffer margins.

These examples also illustrate the heterogeneity of empirical demand distributions across weekday–hour buckets. This reinforces the distinction between buffer construction and stochastic evaluation. The Normal approximation is used only to formulate conservative requirement margins, while the evaluation scenarios are generated empirically from historical observations in order to preserve demand patterns as close as possible to the observed data.

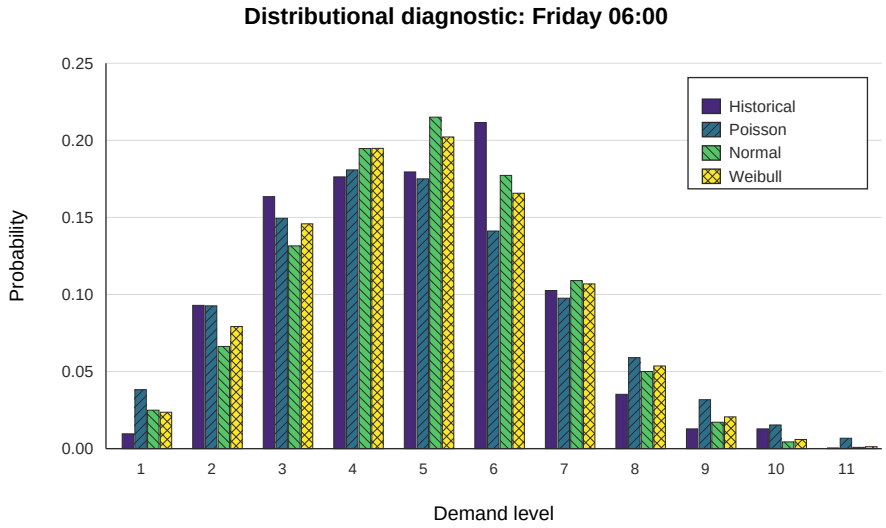

Figure B.1: Distributional diagnostic for Friday 06:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

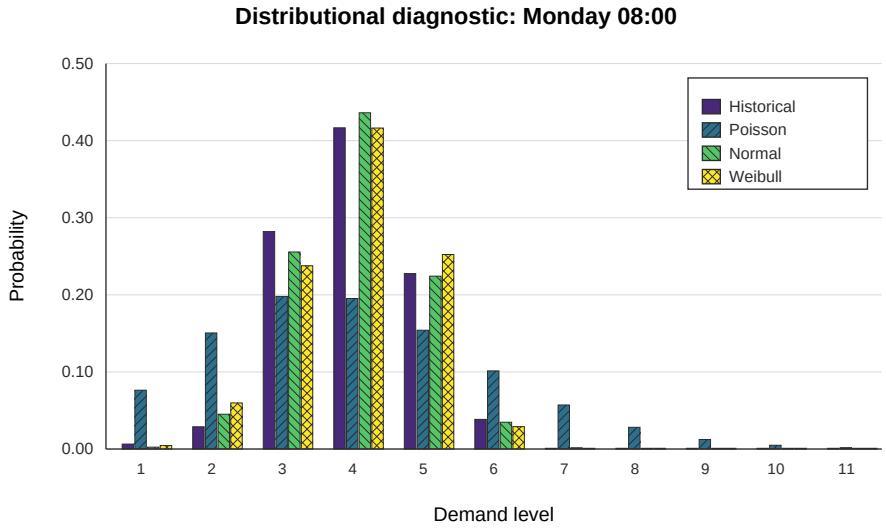

Figure B.2: Distributional diagnostic for Monday 08:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

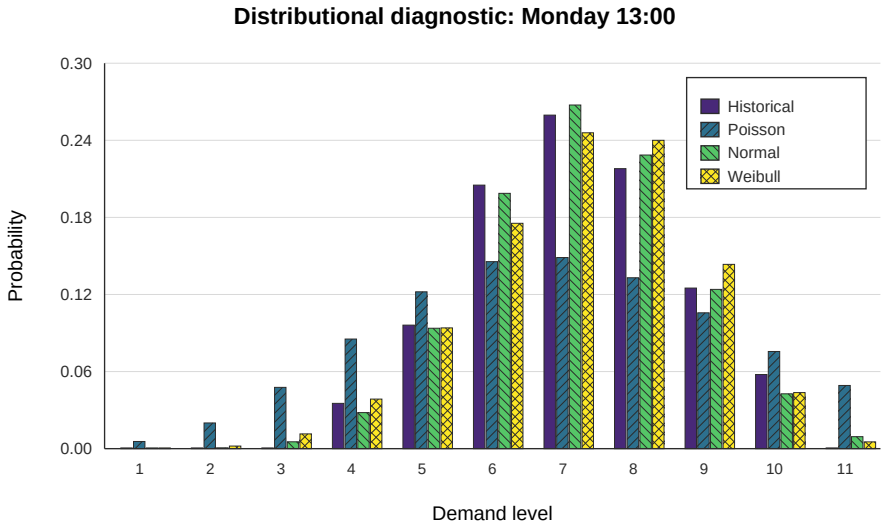

Figure B.3: Distributional diagnostic for Monday 13:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

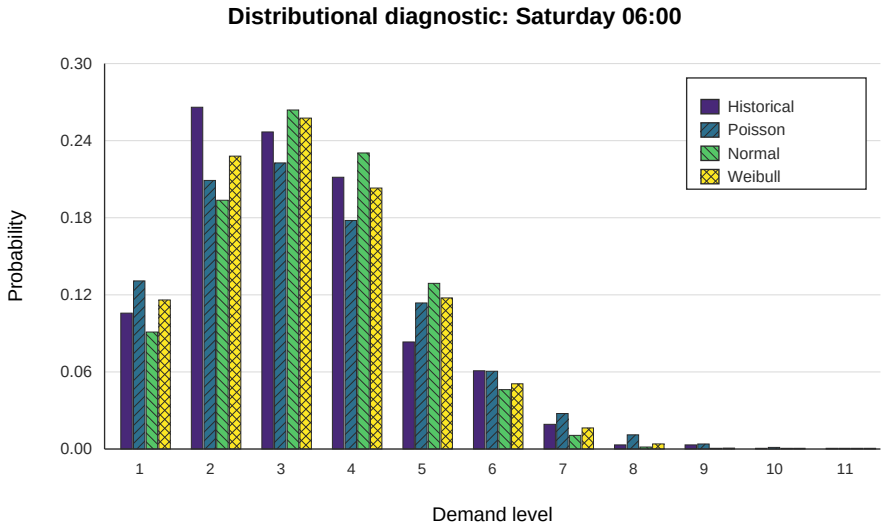

Figure B.4: Distributional diagnostic for Saturday 06:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

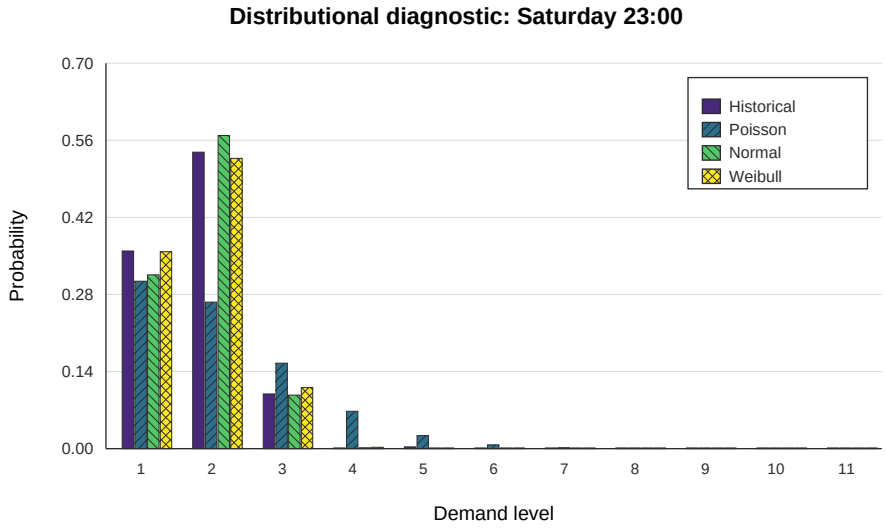

Figure B.5: Distributional diagnostic for Saturday 23:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

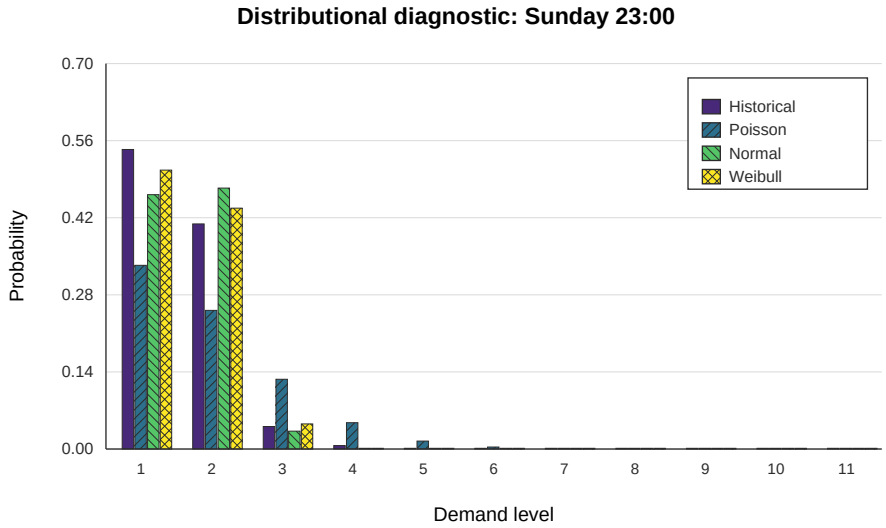

Figure B.6: Distributional diagnostic for Sunday 23:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

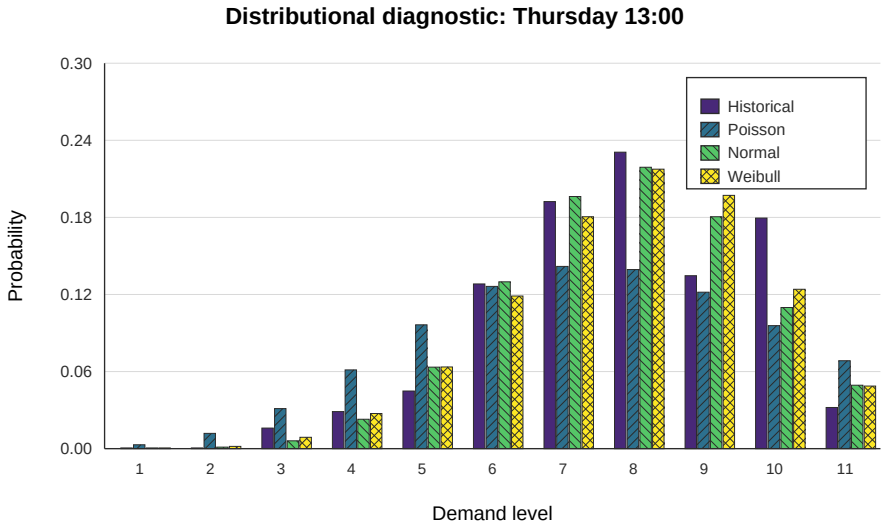

Figure B.7: Distributional diagnostic for Thursday 13:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

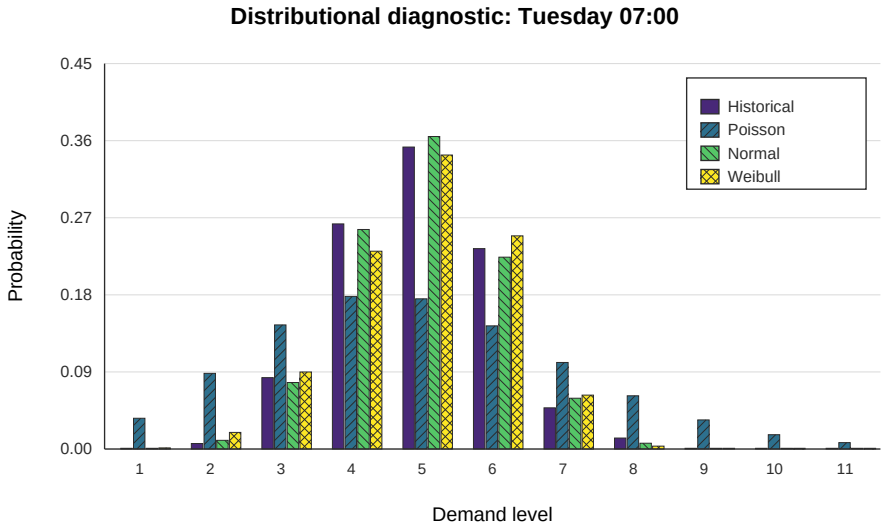

Figure B.8: Distributional diagnostic for Tuesday 07:00. The empirical historical demand distribution is compared with Poisson, Normal and Weibull approximations.

Appendix C

Per-window Results by TCC Level

Table C.1: Per-window calibration of the global buffer, by constrainedness level: the value selected by grid search (Δ^*_{grid}) and by Bayesian optimisation (Δ_{BO}), with the corresponding calibration objective $\widehat{J}$ (training missing hours). Bayesian optimisation is reported here as a calibration-stage diagnostic only. The summary row gives the median of Δ and the mean of $\widehat{J}$.

	TCC 0.7				TCC 0.5			
Window	Δ^*_{grid}	$\widehat{J}_{\text{grid}}$	Δ_{BO}	$\widehat{J}_{\text{BO}}$	Δ^*_{grid}	$\widehat{J}_{\text{grid}}$	Δ_{BO}	$\widehat{J}_{\text{BO}}$
0	0.75	21.8	0.60	24.1	0.99	3.6	0.94	4.0
1	0.6	34.3	0.79	33.7	0.99	5.0	0.73	5.6
2	0.9	25.2	0.79	25.1	0.9	4.2	0.37	7.1
3	0.75	25.7	0.79	25.4	0.99	4.5	0.79	4.6
4	0.3	28.1	0.72	28.2	0.975	4.7	0.73	5.5
5	0.75	22.5	0.65	20.9	0.975	3.4	0.25	4.4
Summary	0.75	26.2	0.75	26.2	0.9825	4.2	0.73	5.2

Table C.2: Per-window out-of-sample performance at TCC 0.7 on S_{test}: baseline and calibrated global buffer (grid). Improvement is the per-window improvement.

	Baseline			Global (grid)			
Window	Mean	P95	Worst	Mean	P95	Worst	Improv. (%)
0	40.4	44.2	48.8	29.2	32.7	36.0	27.7
1	44.3	48.5	51.7	43.5	47.5	51.8	1.7
2	34.5	38.5	40.7	31.0	34.5	37.3	10.1
3	40.8	44.8	49.0	32.2	36.0	41.0	21.2
4	36.5	40.2	42.0	34.5	38.0	40.5	5.5
5	31.0	34.7	37.3	29.6	33.3	35.8	4.3
Mean	37.9	41.8	44.9	33.3	37.0	40.4	11.7

Table C.3: Per-window out-of-sample performance at TCC 0.5 on S_{test}: baseline and calibrated global buffer (grid). Improvement is the per-window improvement.

	Baseline			Global (grid)			
Window	Mean	P95	Worst	Mean	P95	Worst	Improv. (%)
0	7.3	9.0	10.3	4.5	5.8	6.8	39.3
1	12.4	14.2	16.3	5.6	7.2	9.0	54.4
2	11.2	13.0	14.8	4.0	5.3	7.0	64.5
3	8.3	10.0	12.2	4.4	5.7	7.7	47.3
4	8.0	9.8	11.0	4.4	5.8	7.2	45.1
5	7.5	9.2	10.3	4.2	5.7	6.7	43.9
Mean	9.1	10.9	12.5	4.5	5.9	7.4	49.1

Table C.4: Per-window calibration of the block-specific buffer by Bayesian optimisation: the selected per-block conservatism levels $\Delta_1, \ldots, \Delta_5$ and the corresponding calibration objective $\widehat{J}$ (training missing hours). The summary row gives the mean of $\widehat{J}$.

TCC	Window	Δ_1	Δ_2	Δ_3	Δ_4	Δ_5	$\widehat{J}$
0.7	0	0.47	0.98	0.40	0.82	0.80	22.2
	1	0.91	0.89	0.77	0.18	0.80	32.4
	2	0.63	0.68	0.53	0.45	0.55	25.2
	3	0.74	0.86	0.89	0.58	0.80	26.0
	4	0.04	0.71	0.11	0.44	0.20	27.6
	5	0.62	0.00	0.24	0.18	0.77	20.7
	Mean						25.7
0.5	0	0.27	0.98	0.41	0.03	0.34	3.7
	1	0.91	0.41	0.63	0.77	0.97	5.5
	2	0.70	0.85	0.86	0.41	0.89	4.2
	3	0.59	0.98	0.49	0.91	0.43	4.9
	4	0.78	0.62	0.93	0.65	0.91	4.5
	5	0.69	0.59	0.19	0.83	0.35	3.3
	Mean						4.4

Table C.5: Per-window out-of-sample performance of the block-specific buffer at TCC 0.7 on S_{test}. Improvements are per-window, against the baseline and against the global buffer (Table C.2).

	Block (BO)			Improvement (%)	
Window	Mean	P95	Worst	vs baseline	vs global
0	29.9	33.8	35.5	+26.0	−2.4
1	40.8	44.7	48.0	+7.9	+6.3
2	30.9	34.7	37.5	+10.2	+0.1
3	32.4	36.2	40.7	+20.6	−0.7
4	35.0	38.5	41.7	+4.0	−1.6
5	28.0	31.3	35.2	+9.5	+5.4
Mean	32.8	36.5	39.8	+13.0	+1.2

Table C.6: Per-window out-of-sample performance of the block-specific buffer at TCC 0.5 on S_{test}. Improvements are per-window, against the baseline and against the global buffer (Table C.3).

	Block (BO)			Improvement (%)	
Window	Mean	P95	Worst	vs baseline	vs global
0	4.8	6.2	6.8	+34.7	−7.4
1	6.2	8.0	9.8	+49.9	−9.9
2	4.6	5.8	7.8	+59.1	−15.2
3	5.5	7.0	8.5	+34.1	−24.8
4	4.5	6.0	7.0	+43.9	−2.0
5	4.1	5.5	6.3	+45.3	+2.4
Mean	4.9	6.4	7.7	+44.5	−9.5

Appendix D

Solver Diagnostics and Optimality Gaps

Table D.1: Per-window solver diagnostics at TCC 0.7: optimality gap and absolute room (incumbent objective minus lower bound, in worker-hours) for the baseline, the grid-selected global buffer, and the block-specific buffer. The column Δ^* is the global buffer's selected level; the block buffer's per-block levels are listed in Appendix C. The room is small for all three schedules in every window, so the comparisons are not driven by unproven gaps.

		Baseline		Global buffer (grid)		Block buffer (BO)	
Window	Δ^*	Gap%	Room (h)	Gap%	Room (h)	Gap%	Room (h)
0	0.75	0.0	0.00	4.3	0.60	7.1	1.02
1	0.6	20.0	0.17	11.7	1.97	4.6	1.51
2	0.9	5.3	0.12	2.4	1.17	3.9	0.50
3	0.75	28.4	0.33	10.3	1.86	8.7	1.83
4	0.3	4.9	0.32	8.3	1.17	9.2	1.00
5	0.75	5.5	0.16	6.6	2.20	3.8	1.12
Median/max	–	5.4	0.33	7.5	2.20	5.9	1.83

D.1 Incumbent stabilisation and the comparability of time budgets

The robustness argument of Section 6.4.5 rests on the incumbent stabilising well before the time limit, so that the shorter budget used for the block-specific search captures essentially the same schedule as the longer budget used for the grid and global searches. Table D.3 reports, across the 132 grid and global solves (windows 0 to 5, both constrainedness levels, all grid values of Δ), the time for the incumbent objective to come within 0.5 worker-hours of its final value. The median is 186 seconds and the 95th percentile 781 seconds; by 900 seconds the incumbent has stabilised in 96 percent of solves. A small number of solves at near-boundary conservatism levels stabilise later, the latest at about 5800 seconds, but these are exceptions and do not affect the selected configurations.

Table D.2: Per-window solver diagnostics at TCC 0.5: optimality gap and absolute room (incumbent objective minus lower bound, in worker-hours) for the baseline, the grid-selected global buffer, and the block-specific buffer. The column Δ^* is the global buffer's selected level; the block buffer's per-block levels are listed in Appendix C. At this level the block buffer's percentage gaps are larger but rest on near-zero objectives, so the absolute room remains negligible and the comparisons are not driven by unproven gaps.

		Baseline		Global buffer (grid)		Block buffer (BO)	
Window	Δ^*	Gap%	Room (h)	Gap%	Room (h)	Gap%	Room (h)
0	0.99	3.3	0.00	0.0	0.00	3.3	0.00
1	0.99	3.3	0.00	16.6	1.25	25.0	0.17
2	0.9	6.7	0.00	8.3	0.08	0.0	0.00
3	0.99	11.5	0.00	28.5	0.90	16.7	0.00
4	0.975	18.9	0.00	7.7	0.79	10.0	0.33
5	0.975	16.7	0.00	31.1	1.19	33.3	0.00
Median/max	–	9.1	0.00	12.4	1.25	13.4	0.33

Table D.3: Time for the incumbent objective to stabilise, to within 0.5 worker-hours of its final value, across the 132 grid and global solves (windows 0 to 5, both constrainedness levels, all grid values of Δ). The remaining solve time is spent proving optimality rather than improving the schedule.

Elapsed time (s)	Solves with incumbent stabilised (%)
60	20
120	31
300	71
600	87
900	96
1200	97

Appendix E

The non-anticipative ceiling: a sample-average benchmark

This appendix describes the benchmark used in Section 6.4.4 to judge how much room remains above the calibrated buffer, reports its full results, and states what the comparison does and does not establish.

E.1 Purpose and hypothesis

The demand buffer reshapes a fixed coverage budget to reduce missing coverage, but it does not by itself reveal how much of the achievable reduction it captures. The benchmark supplies that reference. For each window and constrainedness level it computes the single fixed roster that minimises expected missing coverage against the training demand distribution, and evaluates it on the test scenarios. Because it commits to one roster before demand is realised, it is non-anticipative in exactly the sense the buffered schedules are, and it is therefore an upper bound on what any non-anticipative schedule, buffered or not, can achieve against that distribution.

The comparison is framed as a one-sided test. Writing E_{buf} for the calibrated buffer's expected missing coverage and E_{saa} for the benchmark's:

- H_0: $E_{\text{buf}} \approx E_{\text{saa}}$. The buffer is already at the non-anticipative ceiling, and no buffer refinement, finer blocks included, can help.
- H_1: $E_{\text{buf}} > E_{\text{saa}}$. Room remains for a better non-anticipative schedule.

The test is informative in one direction only. A null result would be decisive: it would show that refining the buffer is futile. A rejection establishes that room exists, but not that any buffer captures it, because the benchmark is an unconstrained roster rather than a buffered one, and because it is fitted to realised historical demand rather than to the planned requirement the buffer reshapes.

E.2 Method

The benchmark replaces the deterministic requirement with the training demand distribution directly. For each period, the expected missing coverage is a convex, piecewise-linear function of the integer coverage level at that period (Eq. (4.5)), and this function is precomputed from the training scenarios. The scheduling model then minimises the sum of these per-period functions over the planning week, subject to the same workforce size and labour constraints as every other

schedule in this work, using a piecewise-linear objective. This formulation avoids replicating the decision variables across the 500 scenarios and keeps the model the same size as the deterministic one. The coverage budget is fixed at 2904 worker-periods, as everywhere else, and each instance is solved single-threaded under the time limit of Section 5.7. The resulting roster is evaluated on the held-out test scenarios with the procedure used for the baseline and the buffered schedules. Two features matter for interpretation: the benchmark uses no requirement profile, seeing only realised training demand, and it is non-anticipative, producing one fixed roster with no per-scenario recourse.

E.3 Results

Table E.1 reports the benchmark against the baseline and the two buffers, per window and constrainedness level. The benchmark attains less missing coverage than either buffer in every window. Averaged over windows it reduces the baseline shortfall by 29.8% at $k = 0.7$ and 73.1% at $k = 0.5$, against 11.7% and 49.1% for the global buffer and 13.0% and 44.5% for the block buffer. Its optimality gaps are small at $k = 0.7$; at $k = 0.5$ they are larger in relative terms but rest on a near-zero objective (at most 6.9% of a sub-two-hour value), so the reported figures are, if anything, conservative as a ceiling.

Table E.1: Sample-average benchmark against the baseline and the two buffers, per window and constrainedness level: out-of-sample mean missing hours on the test scenarios, with the benchmark's optimality gap.

TCC	Window	Baseline	Global	Block	Benchmark	Gap (%)
0.7	0	40.4	29.2	29.9	22.8	1.4
	1	44.3	43.5	40.8	34.4	0.6
	2	34.5	31.0	30.9	25.1	0.6
	3	40.8	32.2	32.4	26.5	1.0
	4	36.5	34.5	35.0	29.2	0.4
	5	31.0	29.7	28.0	21.5	1.7
	Mean	37.9	33.3	32.8	26.6	
0.5	0	7.3	4.5	4.8	1.9	2.2
	1	12.4	5.6	6.2	3.3	3.0
	2	11.2	4.0	4.6	2.3	0.7
	3	8.3	4.4	5.5	2.2	5.2
	4	8.0	4.4	4.5	3.3	1.6
	5	7.6	4.2	4.1	1.6	6.9
	Mean	9.1	4.5	4.9	2.4	

E.4 What the comparison establishes

Three conclusions follow, and a caveat. First, room remains above the buffer in both regimes: the buffer is far from the best non-anticipative schedule the same data permit. Second, this revises the residual identified in Section 6.4.2: since the benchmark is non-anticipative and still reaches well below the buffer, the residual above the buffer at $k = 0.5$ is not the price of non-anticipativity but

the distance between the buffer-induced roster and the distribution-optimal fixed roster. Third, by the one-sided logic above, the existence of room does not establish that a finer or better-calibrated buffer would capture it; that question is left open.

The caveat concerns information. The benchmark is fitted to realised historical demand and discards the planned requirement, whereas the buffer keeps that requirement and only reshapes it. The room reported here is therefore room under historical information. Whether it would persist against genuine future demand, where the planned requirement may carry forward-looking information the historical record lacks, is beyond what this evaluation can decide; it is discussed in Chapter 7.