



**Università
di Genova**

Università degli Studi di Genova

Department of Mechanical, Energy, Management and
Transportation Engineering (DIME)

Laurea Magistrale in
SAFETY ENGINEERING FOR TRANSPORT, LOGISTICS AND PRODUCTION

**Managing and Optimising Empty-Container Flows
across Multi-Port Multimodal Logistics Networks Using AI
Forecasting (Machine Learning)**

Shokoufeh Yousefi

Supervisor:

Prof. Davide Giglio

Academic Year 2025/2026

July 2026

Abstract

Empty container repositioning is one of the most persistent operational challenges in modern maritime logistics. Global container trade is structurally asymmetric: import-dominated regions accumulate idle equipment, while export-oriented regions face chronic shortages. As a result, roughly one in every five container movements worldwide carries no freight. These empty flows generate no revenue, yet impose substantial cost, congest yard space, and contribute meaningfully to the carbon footprint of global supply chains. The rising urgency of maritime decarbonisation and the growing volatility of post-pandemic freight markets have made the efficient management of empty containers a central concern for shipping lines, port operators, and policymakers alike.

This thesis develops an integrated decision-support framework that couples data-driven machine learning forecasting with risk-aware stochastic optimisation for sustainable and resilient empty container repositioning across multi-port multimodal networks, and delivers it as a working *decision-support platform* that operationalises the framework as software. The framework is organised around two operational steps. The first step forecasts future empty container demand using eight algorithms organised into three families: three classical baselines (Naïve 3M, Drift, Seasonal Naïve) that set the performance floor, four tree-based machine learning models (Random Forest, XGBoost, LightGBM, CatBoost), and one inverse-MAE weighted ensemble that combines the individual learners. Each algorithm is trained independently for every port–container segment, with no cross-segment information sharing. Forecast accuracy is assessed under two complementary protocols: a single-window *back-testing* benchmark and a 156-fold *walk-forward* expanding-window procedure that mimics realistic operational deployment by retraining the models at every planning cycle. The best-performing algorithm is selected separately for each port and container size, with pairwise Diebold–Mariano tests confirming that accuracy differences are not sampling artefacts.

The second step translates the forecasts into actionable repositioning decisions through a two-stage stochastic linear programme with Conditional Value-at-Risk (CVaR) control across road, rail, and short-sea shipping. The model determines how many containers should be moved along each origin–destination corridor and which mode to use, balancing operational cost, transit time, and CO₂ emissions under three objective configurations (cost-focus, balanced, green-focus). The methodological contribution lies in the coupling mechanism itself: demand scenarios are constructed directly from the empirical walk-forward forecast residuals rather than from parametric distributional assumptions, so the optimisation responds to the errors the forecasting model actually makes rather than to a stylised proxy for them. The accompanying decision-support platform implements the framework as a working prototype whose dual-branch architecture executes the back-testing and walk-forward pipelines in parallel, supports manual override

of demand estimates at the planning stage, and exposes a battery of risk diagnostics through a structured tabbed interface.

The framework and its platform are evaluated on a calibrated five-port Italian network covering the Western Mediterranean and Northern Adriatic (Genova, La Spezia, Livorno, Trieste, and Gioia Tauro), using 192 months of monthly data from January 2010 through December 2025. Across the computational experiments, the framework lifts service level from 94.1% under the deterministic baseline to 99.5% under the machine-learning-driven configuration. Demand dynamics vary markedly across ports and container sizes, and no single forecasting model dominates the network; segment-specific selection therefore outperforms a one-size-fits-all approach. A green-focused configuration cuts CO₂ emissions by approximately 18% relative to the balanced-cost configuration at a routing premium of about 9%, quantifying the decarbonisation trade-off. Adding the CVaR constraint raises expected cost by a further 12% but bounds tail-risk under demand uncertainty. The most counterintuitive finding concerns forecast accuracy itself: walk-forward validation nearly triples mean MAE relative to back-testing, yet first-stage cost barely moves. The two-stage recourse channel absorbs the additional forecast-error volume into reshaped second-stage flows, so gains from better forecasting are realised mainly in service-level reliability rather than in nominal cost. Stochastic optimisation thus serves primarily as a robustness mechanism, with value rising as demand uncertainty grows.

Keywords: Empty Container Repositioning; Machine Learning Forecasting; Two-Stage Stochastic Programming; Conditional Value-at-Risk; Multimodal Logistics Optimisation; Sustainable and Resilient Ports; Decision-Support Framework and Platform.

Sommario

Il riposizionamento dei container vuoti rappresenta una delle sfide operative più persistenti nella moderna logistica marittima. Il commercio globale di container è strutturalmente asimmetrico: le regioni a prevalenza di importazioni accumulano attrezzature inutilizzate, mentre le regioni a vocazione esportatrice affrontano carenze croniche. Di conseguenza, circa uno su cinque movimenti di container nel mondo non trasporta alcuna merce. Questi flussi vuoti non generano ricavi, ma comportano costi significativi, congestionano gli spazi di stoccaggio e contribuiscono in modo rilevante all'impronta di carbonio delle catene di approvvigionamento globali. La crescente urgenza della decarbonizzazione marittima e la maggiore volatilità dei mercati merci post-pandemici hanno reso la gestione efficiente dei container vuoti una preoccupazione centrale per le compagnie di navigazione, gli operatori portuali e i decisori politici.

Questa tesi sviluppa un framework integrato di supporto alle decisioni che accoppia la previsione basata sull'apprendimento automatico con l'ottimizzazione stocastica consapevole del rischio per un riposizionamento dei container vuoti sostenibile e resiliente su reti multimodali multi-porto, e lo realizza come una *piattaforma di supporto alle decisioni* funzionante che operationalizza il framework come software. Il framework è organizzato in due fasi operative. La prima fase prevede la domanda futura di container vuoti utilizzando otto algoritmi organizzati in tre famiglie: tre modelli di base classici (Naïve 3M, Drift, Seasonal Naïve), quattro modelli di apprendimento automatico basati su alberi decisionali (Random Forest, XGBoost, LightGBM, CatBoost) e un ensemble pesato per inverso del MAE. Ciascun algoritmo viene addestrato in modo indipendente per ogni segmento porto-container. L'accuratezza di previsione viene valutata mediante due procedure complementari: una valutazione *back-testing* a finestra singola e una procedura *walk-forward* di cross-validation a 156 fold, che simula realisticamente l'implementazione operativa riaddestrando i modelli a ogni ciclo di pianificazione. Il modello con le migliori prestazioni viene selezionato separatamente per ogni porto e dimensione di container, con test di Diebold–Mariano a coppie che confermano che le differenze di accuratezza non sono artefatti di campionamento.

La seconda fase traduce le previsioni in decisioni operative di riposizionamento attraverso un modello di programmazione lineare stocastica a due stadi con controllo del rischio mediante Conditional Value-at-Risk (CVaR), considerando i modi di trasporto stradale, ferroviario e marittimo a corto raggio. Il modello determina quanti container debbano essere movimentati lungo ciascun corridoio origine–destinazione e quale modo utilizzare, bilanciando costi operativi, tempi di transito ed emissioni di CO₂ sotto tre configurazioni dell'obiettivo (cost-focus, balanced, green-focus). Il contributo metodologico principale risiede nel meccanismo di accoppiamento stesso: gli scenari di domanda vengono costruiti direttamente dai residui empirici di previsione walk-forward anziché da ipotesi distributive parametriche. La piattaforma di supporto alle decisioni che accompagna il framework lo implementa come prototipo funzionante la

cui architettura a doppio ramo esegue in parallelo le pipeline back-testing e walk-forward, supporta la modifica manuale delle stime di domanda nella fase di pianificazione, ed espone una serie di diagnostiche di rischio e sensitività attraverso un'interfaccia strutturata a schede.

Il framework e la sua piattaforma sono valutati su una rete calibrata di cinque porti italiani rappresentativi del corridoio del Mediterraneo Occidentale e dell'Adriatico Settentrionale (Genova, La Spezia, Livorno, Trieste e Gioia Tauro), utilizzando 192 mesi di dati mensili da gennaio 2010 a dicembre 2025. Negli esperimenti computazionali, il framework innalza il livello di servizio dal 94,1% del riferimento deterministico al 99,5% nella configurazione guidata dall'apprendimento automatico. Le dinamiche della domanda variano in modo significativo tra porti e dimensioni di container, e nessun singolo modello di previsione domina la rete; la selezione segment-specifica supera quindi un approccio uniforme. Una configurazione green riduce le emissioni di CO₂ di circa il 18% rispetto alla configurazione balanced con un incremento di costo di circa il 9%, quantificando il compromesso della decarbonizzazione. L'aggiunta del vincolo CVaR aumenta il costo atteso di un ulteriore 12% ma limita il rischio di coda sotto incertezza della domanda. Il risultato più controintuitivo riguarda l'accuratezza di previsione stessa: la validazione walk-forward triplica quasi il MAE medio rispetto al back-testing, eppure il costo del primo stadio si muove a malapena. Il canale di recourse del secondo stadio assorbe il volume aggiuntivo di errore di previsione nei flussi del secondo stadio riformulati, cosicché i guadagni derivanti da previsioni migliori si realizzano principalmente in termini di affidabilità del livello di servizio piuttosto che in costo nominale. L'ottimizzazione stocastica funge quindi principalmente da meccanismo di robustezza, con valore crescente al crescere dell'incertezza della domanda.

Parole chiave: Riposizionamento di Container Vuoti; Previsione mediante Apprendimento Automatico; Programmazione Stocastica a Due Stadi; Conditional Value-at-Risk; Ottimizzazione Logistica Multimodale; Porti Sostenibili e Resilienti; Framework e Piattaforma di Supporto alle Decisioni.

Acknowledgements

The completion of this thesis would not have been possible without the guidance, support, and encouragement of several people to whom I am deeply grateful.

First and foremost, I would like to express my sincere gratitude to my advisor, **Prof. Davide Giglio**, for his invaluable guidance, constructive feedback, and continuous support throughout the development of this work. His expertise in transport engineering and stochastic optimisation has shaped the direction of this thesis in essential ways, and his patience and availability have been a constant source of motivation. It has been a privilege to work under his supervision.

I am also profoundly thankful to my husband, **Dr. Hamid Livani**, a PhD candidate at the University of Genova, whose support throughout this journey has been beyond measure. His scientific insight and willingness to discuss ideas at any hour have enriched both the technical depth and the clarity of this work. Equally important has been his unwavering emotional support during the most demanding phases of this research. Without his patience, encouragement, and belief in me, this thesis would not have reached its current form.

I would also like to express my deepest gratitude to my family, whose love and sacrifices have always accompanied me, even from far away. During the difficult circumstances and the painful times of war in Iran, their greatest concern was still my success, my safety, and my future. Their strength, patience, and unconditional support have been a constant source of courage for me. I carry their love with me in every step of this journey.

To my advisor, my husband, and my family, I offer my heartfelt thanks.

Shokoufeh Yousefi
Genova, July 2026

Contents

Abstract

Sommario

Acknowledgements

1	Introduction	6
1.1	Background and Motivation	6
1.1.1	Maritime transport and the rise of containerisation	6
1.1.2	The empty-container repositioning problem	6
1.1.3	The Italian and Mediterranean context	7
1.1.4	Economic and environmental stakes	7
1.1.5	Two parallel research streams	8
1.1.6	The policy backdrop	8
1.1.7	Positioning and conceptual framework	8
1.2	Problem Statement	9
1.3	Research Questions	10
1.4	Research Hypotheses	11
1.5	Research Objectives	11
1.6	Contributions	12
1.7	Scope and Limitations	14
1.8	Thesis Structure	15
2	Literature Review	17
2.1	Empty container logistics and the repositioning problem	17
2.2	Traditional forecasting methods in maritime logistics	18
2.3	Machine learning for container demand forecasting	19
2.4	Optimisation models for empty container repositioning	20
2.5	Artificial intelligence in port operations	21
2.6	Integrated approaches and the research gap	22
2.7	Chapter summary	24

3	Methodology	25
3.1	Framework overview	25
3.2	Data and study area	27
3.2.1	Port network and coverage	27
3.2.2	Panel structure and dataset	27
3.2.3	Temporal coverage and operating regimes	28
3.2.4	Data construction and calibration boundary	28
3.2.5	Quality pipeline	29
3.2.6	Data ingest in the decision-support platform	29
3.3	Forecasting framework	30
3.3.1	Forecasting model portfolio	30
3.3.2	Evaluation protocols	32
3.3.3	Forecasting in the decision-support platform	34
3.3.4	Accuracy metrics	37
3.3.5	Pairwise model comparison: Diebold–Mariano test	38
3.3.6	Best-model selection rule	39
3.3.7	Forecast analysis in the decision-support platform	39
3.4	Port-specific demand estimation	42
3.4.1	Point demand forecast	42
3.4.2	Empirical residual sampling	43
3.4.3	Demand review in the decision-support platform	45
3.5	Stock initialisation	46
3.6	Supply-demand balance	48
3.7	Multimodal transport network	49
3.7.1	Network topology	49
3.7.2	Per-mode unit parameters	50
3.7.3	Arc capacities	51
3.7.4	Representative corridor: Genova–La Spezia	51
3.7.5	Network configuration in the decision-support platform	52
3.8	Two-stage stochastic optimisation model	53
3.8.1	Sets, indices, and parameters	54
3.8.2	Route-aware recourse penalty	54
3.8.3	Multi-objective configuration of arc costs	55
3.8.4	Decision variables	56
3.8.5	Objective function	58
3.8.6	Constraints	59
3.9	Decision-support platform architecture	62
3.9.1	Modular architecture and software stack	62
3.9.2	ten-tab user interface	63
3.9.3	Dual-branch BT/WF execution	63

3.9.4	Optimisation execution	64
3.9.5	Repositioning plan visualisation	66
3.9.6	Experiments, scenarios, and diagnostics	67
3.9.7	Pareto frontier diagnostic	70
3.10	Chapter summary	71
4	Computational Study	73
4.1	Case study network and dataset	73
4.1.1	Geographic configuration and operational roles	74
4.1.2	Multimodal arcs and modal connectivity	74
4.1.3	Port-level inputs to the optimisation	75
4.2	Forecasting experimental design	75
4.2.1	Training and testing splits	75
4.2.2	Walk-forward fold structure and computational cost	76
4.2.3	Hyperparameter selection policy	76
4.3	Optimisation scenario design	77
4.3.1	The six end-to-end scenarios	77
4.3.2	Pairwise comparisons that isolate effects	78
4.4	Robustness analyses	79
4.4.1	Monte Carlo robustness study	80
4.4.2	Scenario-construction-method comparison	80
4.5	Software stack and computational infrastructure	81
4.5.1	Software stack	81
4.5.2	Computational infrastructure	81
4.6	Reproducibility protocol	82
4.7	Chapter summary	83
5	Results	85
5.1	Step 1: Forecasting results	85
5.1.1	Forecasting accuracy on a representative segment	85
5.1.2	Visual comparison of walk-forward forecasts	87
5.1.3	Best model selection across port–container segments	88
5.1.4	Pairwise Diebold–Mariano significance results	90
5.1.5	December 2025 demand forecasts	91
5.2	Step 2: Optimisation results	92
5.2.1	Supply-demand balance	92
5.2.2	Optimised repositioning plan	93
5.2.3	Comparative analysis of the six scenarios	95
5.3	Pareto frontier: cost versus CO ₂ emissions	98
5.3.1	Empirical frontier values	99
5.3.2	Convexity of the frontier	99

5.3.3	Near-coincidence of the BT and WF frontiers	100
5.4	Monte Carlo robustness	101
5.4.1	Variability of expected cost	101
5.4.2	Variability of CVaR	102
5.4.3	Variability of service level	102
5.4.4	Implications for the inter-scenario findings	102
5.5	Scenario-construction-method comparison	103
5.5.1	Empirical versus parametric Gaussian	103
5.5.2	Empirical versus historical bootstrap	104
5.5.3	Synthesis: why empirical residual sampling matters	104
5.6	Chapter summary	105
6	Discussion	107
6.1	Why no single forecasting model wins everywhere	107
6.2	The recourse channel as an error absorber	108
6.3	Bias, variance, and what stochastic programming can fix	110
6.3.1	The distinction in operational terms	110
6.3.2	Implications for forecasting effort	110
6.3.3	Why this distinction matters for the literature	111
6.4	The shape of the cost–emission trade-off	111
6.4.1	The convexity of the frontier	112
6.4.2	Decomposition of the corner-to-corner premium	112
6.4.3	Practical implication: a green-leaning balanced configuration	113
6.5	Three layers of uncertainty, three different jobs	113
6.5.1	What each layer does	113
6.5.2	Why the three layers are non-substitutable	114
6.5.3	When each layer is most valuable	115
6.6	Managerial and policy implications	115
6.6.1	Shipping lines	115
6.6.2	Port operators	116
6.6.3	Inland transport providers	117
6.6.4	Regulators and policy-makers	117
6.7	Boundary conditions and deployment considerations	118
6.7.1	The calibration boundary	118
6.7.2	The single-period horizon	119
6.7.3	Cross-port independence in scenario construction	120
6.7.4	Network specifics and generalisation	120
6.7.5	Live deployment beyond the framework itself	121
6.8	Chapter summary	122

7	Conclusion and Future Work	124
7.1	Revisiting the research questions	124
7.1.1	RQ1: forecasting heterogeneity	125
7.1.2	RQ2: forecast–optimisation coupling	125
7.1.3	RQ3: cost–emission trade-offs	126
7.1.4	RQ4: decision-support platform feasibility	127
7.2	Methodological contributions	128
7.2.1	Integrated forecast-driven stochastic optimisation framework	128
7.2.2	Eight-algorithm cross-paradigm forecasting evaluation	128
7.2.3	The recourse channel as a forecast-error absorber	129
7.2.4	Route-aware recourse penalty formulation	129
7.2.5	Dual-branch BT/WF architecture as a methodological commitment	130
7.3	Applied contributions	130
7.3.1	Working modular decision-support platform	130
7.3.2	Empirical case study on a calibrated five-port Italian network	131
7.3.3	Three layers of uncertainty management	131
7.4	Practical implications	131
7.5	Future work	133
7.5.1	Validation on proprietary carrier data	133
7.5.2	Multi-period rolling-horizon extension	134
7.5.3	Multivariate residual modelling under joint shocks	135
7.6	Closing remarks	136

Chapter 1

Introduction

This chapter sets out the motivation, scope, and overall structure of the thesis. Section 1.1 situates the empty-container repositioning problem within contemporary maritime logistics, and Section 1.2 identifies the operational gap that the work addresses. Three short sections then state the research questions (Section 1.3), the hypotheses (Section 1.4), and the objectives (Section 1.5) that organise the rest of the investigation. Section 1.6 lists the methodological and applied contributions, Section 1.7 marks the boundaries within which they should be read, and Section 1.8 previews the chapters that follow.

1.1 Background and Motivation

1.1.1 Maritime transport and the rise of containerisation

Maritime transport is the structural backbone of global trade. More than 80% of world merchandise by volume moves by sea, and the share carried in containers continues to grow as supply chains become longer and more dependent on standardised intermodal handling (United Nations Conference on Trade and Development, 2023). The expansion of containerised shipping and the deployment of progressively larger vessels have turned container terminals into critical nodes where goods are stored, transhipped, and redistributed before continuing along multimodal supply chains, so the performance of these terminals shapes the reliability of international trade well beyond the port gate.

The shipping container itself, introduced in the late 1950s, made intermodal transfer possible without repacking cargo at every interface. The decades since have brought exponential growth in containerised volumes, the development of dedicated terminals at major ports, and the emergence of the global liner alliances that dominate ocean-going services today. With that growth came a deeper shift in what terminals *are*: they have moved from cargo-handling sites toward automated, capital-intensive nodes whose throughput and turnaround times directly affect the competitiveness of the regions they serve.

1.1.2 The empty-container repositioning problem

A sizeable share of this activity involves containers that carry no freight. Empty units account for roughly 20% of global container movements, a share that persists—and arguably grows—as the gap between

import-dominated and export-oriented regions widens (United Nations Conference on Trade and Development, 2023; Song and Carter, 2009; Karmelić et al., 2012). The cause is structural: when goods flow predominantly in one direction, the containers that carried them must eventually return, and return loads rarely suffice to balance the system. Empty equipment therefore has to be moved continuously from surplus regions to shortage regions. These movements generate no direct revenue, yet they are an operational imperative—exporters cannot ship without equipment, and ports cannot function with yards saturated by idle units (Theofanis and Boile, 2009; Braekers et al., 2011). They also carry an environmental footprint: inefficient empty flows inflate transport emissions, which makes their management an under-recognised lever for maritime decarbonisation.

The difficulty of managing these flows comes from several interacting forces. Structural trade imbalances drive persistent accumulation in some regions and chronic shortage in others (Song and Dong, 2012; Abdelshafie et al., 2023). Demand volatility from seasonal cycles, economic fluctuations, geopolitical disruption, and vessel-schedule irregularities amplifies these baseline pressures. A shortage delays exports, idles vessels, and inflates logistics costs; a surplus saturates yards and inflates storage expenses. Both outcomes are expensive, but they call for different operational responses, and so effective management depends on anticipating imbalances early and coordinating repositioning decisions across the network (Lam, 2014; Sommer et al., 2025).

1.1.3 The Italian and Mediterranean context

The Italian and broader Mediterranean setting sharpens the problem further. Italian ports lie between the manufacturing economies of Northern Europe and the deep-sea lanes that connect Asia, Africa, and the Americas through the Suez Canal. The Western Mediterranean and Northern Adriatic corridors host a heterogeneous mix: gateway ports such as Genova and La Spezia serve Northern Italy and Switzerland, a transshipment hub such as Gioia Tauro feeds the deep-sea network, and regional gateways such as Trieste orient toward Central and Eastern Europe (United Nations Conference on Trade and Development, 2023). The resulting network carries pronounced and persistent imbalances: some ports systematically accumulate empties while others run short, and significant volumes have to be repositioned on a continuous basis (Song and Dong, 2012; Abdelshafie et al., 2023). Optimising those flows matters not only academically but for the competitiveness of Italian logistics and the wider European transport-policy agenda.

1.1.4 Economic and environmental stakes

The stakes are substantial. Repositioning empties consumes fuel, road and rail capacity, and port-handling resources without producing revenue, and it generates non-trivial CO₂ emissions across the maritime chain. Regulatory pressure—through the International Maritime Organization’s revised greenhouse-gas strategy and the European Union’s inclusion of maritime transport in its Emissions Trading System—has turned the reduction of unnecessary empty movements from a cost-saving option into a compliance obligation. The disruptions of the past few years—the COVID-19 pandemic, ongoing geopolitical conflicts, and the consequent rerouting of major shipping lanes—meanwhile exposed how fragile reposition-

ing systems designed for stable demand can be (Notteboom et al., 2021; Toygar et al., 2024). Planning approaches that are simultaneously cost-efficient, low-emission, and resilient under uncertainty are no longer optional (Lam, 2014; Sommer et al., 2025).

1.1.5 Two parallel research streams

Two streams of research have moved in parallel without quite meeting. On the predictive side, advances in artificial intelligence and data-driven analytics now allow forecasting models to capture nonlinear demand patterns and temporal dependencies that conventional statistical methods miss (Jahangard et al., 2025; Xu et al., 2022; Leng et al., 2025). In container throughput and equipment-availability prediction, machine learning has been shown to outperform classical baselines under appropriate conditions (Xu et al., 2022; Kulshrestha et al., 2024; Ratmoko and Zuliarso, 2025). On the prescriptive side, stochastic programming offers a practical way to embed uncertainty inside the optimisation rather than treat it as an afterthought (Vassos et al., 2025). Two-stage formulations are particularly well suited to repositioning, since the first-stage commitment is fixed before demand is fully revealed and a recourse layer absorbs whatever realisation materialises later.

The literature has rarely connected the two streams. Forecasting studies typically chase predictive accuracy and stop there, with little follow-through to the decisions those forecasts are meant to inform (Martius et al., 2022; Gouaboua et al., 2024). Optimisation studies, conversely, treat demand as an exogenous input—a point estimate or a hand-built scenario set—rather than as the output of a calibrated forecasting model (Olivo et al., 2005; Shintani et al., 2007; Song et al., 2022; Abdelshafie et al., 2023; Sommer et al., 2025). Each stream offers strong components; few studies wire them into a single decision pipeline (Jahangard et al., 2025; Zhou et al., 2021). The implications of this separation are developed in Section 1.2 below.

1.1.6 The policy backdrop

A clear policy backdrop frames the work. The International Maritime Organization’s 2023 GHG Strategy (International Maritime Organization, 2023) commits the global shipping sector to substantial emission reductions over the coming decades, and initiatives such as Green and Digital Shipping Corridors and Just-In-Time port-call coordination already target vessel-side decarbonisation. Equivalent landside instruments for empty-equipment management remain underdeveloped, even though the modal shift on which port-level decarbonisation depends is ultimately a landside problem. The thesis addresses this landside gap directly.

1.1.7 Positioning and conceptual framework

The thesis sits at the interface between the two streams of Section 1.1.5. Its contribution is not a new forecasting algorithm or a new optimisation formulation in isolation, but the design of an integrated pipeline—realised both as a methodological *framework* and as a working decision-support *platform*—in which the predictive layer drives the prescriptive layer through scenario construction from forecast

residuals.

Conceptually the framework is a two-stage pipeline. In the first stage, a portfolio of forecasting models competes segment by segment for the most accurate prediction of empty container demand at each port and container type, and the best-performing model on each segment is retained. The output is not a single number but a calibrated forecast together with the residual distribution of the chosen model. That residual distribution is the bridge to the second stage: instead of discarding the residuals or replacing them with a parametric proxy, the framework samples from them directly to construct optimisation scenarios that reflect the forecast’s actual error structure. The scenarios then feed a two-stage stochastic optimisation that decides how many empty containers to send along each origin–destination corridor and which mode—road, rail, or short-sea—to use, balancing cost, transit time, and CO₂ emissions under user-specified weights and a Conditional Value-at-Risk (CVaR) constraint on tail-cost exposure. The output is a repositioning plan that is robust across the scenario distribution and that exposes the genuine trade-offs between cost, service level, and environmental performance.

The framework is operationalised both as a methodology—specified in mathematical and algorithmic detail in Chapter 3—and as a working decision-support platform, documented in the same chapter and illustrated through annotated screenshots of its ten-tab interface.

1.2 Problem Statement

Empty container repositioning has received decades of academic and industrial attention, yet it remains operationally unresolved. The mathematical formulation is not the obstacle: network-flow problems are well understood and solve quickly with off-the-shelf software. The obstacle is the joint effect of demand uncertainty, multimodal trade-offs, and the gap between predictive analytics and operational decision-making. Three weaknesses in current practice motivate the present study.

Reliance on deterministic, point-forecast planning. Traditional repositioning rests on historical averages, deterministic linear programmes, and expert judgement (Olivo et al., 2005; Shintani et al., 2007). These tools are transparent and simple, but poorly equipped to handle uncertainty: a plan that treats demand as known underperforms whenever reality departs from the assumption, which it does regularly. The COVID-19 pandemic and the freight-market turbulence that followed made the operational and financial cost of that mismatch difficult to ignore (Notteboom et al., 2021; Toygar et al., 2024).

Disconnect between forecasting and optimisation. Even when optimisation does account for uncertainty, the scenarios driving it are usually drawn from parametric assumptions—typically a normal distribution fitted to aggregate demand—rather than from the residuals of a calibrated forecasting model (Song et al., 2022; Vassos et al., 2025). The two layers then speak different statistical languages: the optimiser sees an idealised picture of error, not the errors the forecaster actually makes. Risk measures such as Conditional Value-at-Risk become miscalibrated, and the resulting plans carry hidden sensitivity to realisations outside the parametric envelope (Vassos et al., 2025; Cao et al., 2023).

Limited multimodal and emission scope. Many repositioning frameworks consider only road transport, treating mode as exogenous (Olivo et al., 2005; Shintani et al., 2007). In practice, repositioning across an interconnected network involves genuine trade-offs between road, rail, and short-sea shipping, each with distinct cost, time, and emission profiles. Mode selection is a central decision variable, not a side issue, and that becomes especially clear under tightening decarbonisation regulation (Khazzar et al., 2025; Jahangard et al., 2025). A framework that ignores it forfeits operating points that are cheaper, faster, and cleaner at once.

Together these three weaknesses define the gap the thesis addresses: no widely adopted framework simultaneously (i) handles demand uncertainty in a principled way, (ii) derives uncertainty quantification from calibrated forecasting models rather than parametric assumptions, and (iii) makes multimodal trade-offs explicit under cost-, time-, and emission-aware objectives.

Closing this gap is not a purely academic exercise. Port operators and shipping companies face concrete monthly decisions: *how many empty containers should move from Genova to La Spezia next month, and on which mode?* Road is fast but expensive and emission-intensive; rail is cheap and clean but capacity-constrained; sea is the cheapest per TEU but slow and operationally rigid. Under uncertain demand, the right answer depends not only on the expected volume but on the planner’s tolerance for short-age risk. A framework that produces auditable multimodal plans under explicit uncertainty assumptions therefore has direct operational value, particularly as the sector moves toward stricter emission-reporting and decarbonisation targets.

1.3 Research Questions

The thesis is organised around four research questions, investigated through the framework of Chapter 3 and the computational study of Chapter 4.

- **RQ1.** *Which forecasting approach—among classical baselines, tree-based machine learning models, and a weighted ensemble—most accurately predicts empty container demand at the port-container-type level, and how stable is this ranking under operationally realistic out-of-sample evaluation?*
- **RQ2.** *Does grounding stochastic scenario construction in the empirical residuals of a calibrated forecasting model translate into demonstrable operational benefits, and how does forecast accuracy propagate through the optimisation pipeline to influence operating cost and service-level reliability?*
- **RQ3.** *What are the achievable trade-offs between operational cost, service level, and CO₂ emissions when empty container repositioning is jointly optimised across road, rail, and short-sea transport modes, and how do these trade-offs respond to tail-risk control through a CVaR constraint?*
- **RQ4.** *To what extent can the integrated forecast-driven stochastic optimisation pipeline be realised as a working, modular decision-support platform suitable for tactical planning use by port operators and shipping companies?*

The four questions form a logical progression: from the predictive layer (RQ1), through the linkage between prediction and prescription (RQ2), to the prescriptive layer with its sustainability implications (RQ3), and finally to the engineering question of practical deployability (RQ4).

1.4 Research Hypotheses

Each research question is paired with a working hypothesis tested empirically in Chapter 4.

- **H1.** No single forecasting method is uniformly superior across all port–container-type segments. The best-performing model varies with the local demand structure, and the relative ranking may differ between in-sample (back-testing) and out-of-sample (walk-forward) evaluation (Xu et al., 2022; Chan et al., 2019; Henesey and Liu, 2020).
- **H2.** Demand scenarios constructed from empirical walk-forward residuals produce optimisation outcomes operationally aligned with the deployed forecaster’s actual error characteristics. The two-stage recourse channel absorbs forecast-error volume into reshaped second-stage flows, so gains from better forecasting are realised primarily as service-level reliability rather than as proportional reductions in nominal first-stage cost (Vassos et al., 2025; Song et al., 2022).
- **H3.** Multimodal optimisation across road, rail, and short-sea shipping identifies Pareto-efficient operating points that simultaneously reduce cost and CO₂ emissions relative to road-only baselines (Jahangard et al., 2025; Khazzar et al., 2025). The cost–emission frontier is convex: emission reductions in the neighbourhood of the balanced configuration are inexpensive, but become disproportionately costly close to the green-optimal corner.
- **H4.** The complete forecasting–optimisation pipeline can be implemented as a working decision-support platform with a modular software architecture that accepts standard operational inputs (historical port flows, transport network parameters, risk preferences) and produces deployable repositioning plans within operationally acceptable computation times on commodity hardware (Leng et al., 2025).

1.5 Research Objectives

The overarching objective is to design, implement, and empirically evaluate an integrated decision-support framework for sustainable and resilient empty container repositioning across multi-port logistics networks, and to deliver it as a working platform. This decomposes into six specific objectives.

1. **Develop a port- and container-type-specific forecasting layer** that competitively evaluates eight algorithms organised into three families (three classical baselines, four tree-based machine learning models, and one inverse-MAE weighted ensemble) under both single-window back-testing and 156-fold expanding-window walk-forward cross-validation, with pairwise Diebold–Mariano testing of accuracy differences.

2. **Design an empirical scenario-generation mechanism** that constructs demand scenarios directly from walk-forward forecast residuals, preserving the actual asymmetry and tail behaviour of forecast errors without imposing parametric distributional assumptions.
3. **Formulate and solve a two-stage stochastic optimisation model with CVaR control** for empty-container repositioning across road, rail, and short-sea shipping, supporting cost-focused, balanced, and green-focused multi-objective configurations, with a route-aware recourse penalty that reflects the cheapest reachable emergency repositioning at each destination port.
4. **Investigate the operational implications of the forecast–optimisation coupling** through systematic experiments across multiple forecasting protocols and risk preferences, characterising how forecast-error volume is absorbed by the recourse channel and how the resulting gains accrue to service-level reliability rather than to nominal first-stage cost.
5. **Implement the complete pipeline as a modular decision-support platform** in Python, organised as a tabbed interface that exposes each pipeline stage (data ingest, forecasting, scenario generation, optimisation, results, and robustness diagnostics) as an independently auditable module, accompanied by reproducible scripts and validated on a calibrated five-port Italian network spanning 192 months of monthly data.
6. **Evaluate the framework and its platform against operationally meaningful key performance indicators**, including service level, expected cost, CVaR-based tail risk, CO₂ emissions, and modal split, across six representative end-to-end scenarios that progressively layer forecasting intelligence, stochastic optimisation, and CVaR risk control under three objective configurations.

1.6 Contributions

The contributions of the thesis are both methodological and applied.

Methodological contributions.

- (1) **An integrated forecast-driven stochastic optimisation framework** in which the forecasting and optimisation layers are coupled through empirical residual sampling rather than treated as independent sub-problems. The integration is a methodological advance over the prevailing approach in the empty-container literature, where forecasting outputs and optimisation scenarios are typically constructed in disconnected pipelines (Abdelshafie et al., 2023; Sommer et al., 2025; Zugno et al., 2025).
- (2) **A rigorous cross-paradigm evaluation of eight forecasting algorithms** spanning three classical baselines, four tree-based machine learning models (Random Forest, XGBoost, LightGBM, CatBoost), and one inverse-MAE weighted ensemble, applied to all ten port–container-type segments of the case-study network under both back-testing and walk-forward protocols (Xu et al., 2022; Kulshrestha et al., 2024; Chan et al., 2019). Pairwise Diebold–Mariano testing across 280 comparisons

confirms that the observed accuracy differences are not sampling artefacts. To the best of the author's knowledge, this is the most comprehensive empirical comparison of forecasting paradigms reported in the empty-container repositioning context with full statistical-significance treatment.

- (3) **An explicit characterisation of the recourse channel as a forecast-error absorber.** The thesis demonstrates that the two-stage stochastic structure absorbs forecast-error volume into reshaped second-stage flows, so that improvements in forecasting accuracy translate primarily into gains in service-level reliability rather than into proportional reductions in first-stage cost. This mechanism is rarely articulated explicitly in the empty-container literature, and it has direct consequences for the investment case in forecasting infrastructure under different contractual environments (Vassos et al., 2025).
- (4) **A route-aware recourse penalty formulation** in which the cost of unmet demand at each destination port is set equal to the cheapest reachable emergency repositioning into that port plus a service-delay surcharge. This replaces the standard fixed-penalty convention with a port-specific, network-consistent recourse cost that the optimiser cannot game by exploiting artificially low short-age penalties, and that gives a more faithful representation of the operational economics of unmet demand.
- (5) **A clarification of the bias–variance distinction in what stochastic programming can fix.** The recourse-absorption mechanism works on the variance component of forecast error but not on its bias: bias correction and variance reduction look like complements in an accuracy table, yet behave as substitutes in operational terms. The distinction has direct managerial implications for the allocation of forecasting effort under stochastic repositioning, and is rarely made explicit in the existing ECR literature.

Applied contributions.

- (6) **A working, modular decision-support platform** implemented in Python and organised as a tabbed interface in which each pipeline stage (data ingest, forecasting, scenario generation, optimisation, results, and robustness diagnostics) is exposed as an independently auditable module. The platform's dual-branch architecture runs the back-testing and walk-forward pipelines in parallel and surfaces both estimates side by side, so that the operationally honest figure remains visible at every step of the analysis rather than buried in a single summary statistic. The platform is designed to integrate with existing port and carrier ERP environments rather than replace them (Jahangard et al., 2025; Khazzar et al., 2025).
- (7) **An empirical case study on a calibrated five-port Italian Mediterranean network** (Genova, La Spezia, Livorno, Trieste, and Gioia Tauro) using 192 months of monthly data spanning three structurally distinct demand regimes: a pre-disruption baseline (2010–2019), the COVID-19 shock and recovery (2020–2022), and the post-pandemic period of geopolitical freight volatility (2023–2025) (United Nations Conference on Trade and Development, 2023; Notteboom et al., 2021; Toygar

et al., 2024). The results provide quantitative evidence that AI-driven multimodal stochastic optimisation can simultaneously improve service level (from 94.1% to 99.5%), constrain tail-risk through CVaR, and reduce CO₂ emissions (by approximately 18% at a routing premium of about 9%) relative to a deterministic baseline.

- (8) **An operational characterisation of three layers of uncertainty management.** The framework treats demand uncertainty at three non-substitutable points: machine learning forecasting reduces average error, two-stage recourse absorbs realised error by reshaping second-stage flows, and the CVaR constraint bounds tail error in the worst 5% of demand realisations. The pairwise comparison isolating the CVaR effect shows that the constraint accepts a moderate increase in expected cost in exchange for reducing exposure to extreme realisations—a robustness premium whose operational relevance grows sharply under high-impact disruptions.

1.7 Scope and Limitations

The scope of the thesis is bounded along several dimensions.

Geographic scope. The case study covers a five-port Italian network in the Western Mediterranean and Northern Adriatic. These ports together handle more than 60% of Italian containerised trade and span a representative mix of surplus and deficit roles, but the specific quantitative results should not be assumed to generalise mechanically to networks with different structural characteristics, regulatory environments, or modal mixes.

Temporal scope. The dataset spans 192 months from January 2010 through December 2025, covering the pre-pandemic baseline (2010–2019), the COVID-19 disruption (2020–2022), and the post-pandemic period of geopolitical freight volatility (2023–2025). Planning is performed at monthly granularity, which matches the cadence used by many shipping lines and depot operators but does not capture intra-month dynamics such as vessel-call clustering or daily yard congestion.

Modelling scope. The thesis focuses on the repositioning problem in isolation; it does not jointly optimise over related decisions such as vessel scheduling, container leasing, or detention-and-demurrage management. Containers are modelled as two aggregated categories (20DV and 40HC), which captures the dominant operational distinctions but abstracts away from specialised equipment. The opening supply of empty containers at each port is treated as a known, non-stochastic input; second-stage decisions concern only how to redistribute that supply. The framework is formulated as a single-period two-stage stochastic programme, and inventory carry-over between consecutive planning months is left for future multi-period extensions.

Data scope. Port-level empty-container flows disaggregated by container type are commercially sensitive and not consistently published (Song and Carter, 2009; Vassos et al., 2025; Sommer et al., 2025).

The dataset used here is therefore constructed through a calibrated procedure anchored to public aggregate statistics from Assoporti and Eurostat. Working with calibrated rather than proprietary data is established practice in stochastic logistics research when carrier-level records are inaccessible; the contribution of such studies rests on the modelling framework and the structural insights it produces rather than on the confidentiality of the underlying figures. Absolute cost and emission values should therefore be read as conditional on the calibration boundary, while the methodological contributions and the relative comparisons across configurations do not depend on the precise absolute values.

Implementation scope. The decision-support platform is delivered as a working research-grade prototype rather than as a production-hardened product. User authentication, persistent storage, real-time data ingestion, and production-grade GUI design lie outside the scope, though the modular architecture has been chosen to support extension along these lines.

1.8 Thesis Structure

The remainder of the thesis is organised into six chapters, briefly summarised below.

Chapter 2 – Literature Review. Reviews the academic and industrial literature relevant to the study across six thematic areas: empty container logistics and the repositioning problem; classical forecasting methods in maritime logistics; machine learning for container demand forecasting; optimisation models for empty container repositioning; artificial intelligence in port operations; and integrated approaches and the research gap. The chapter maps the methodological coverage of representative ECR and related frameworks against the framework proposed here, and articulates the four specific gaps that motivate the methodological contributions of the thesis.

Chapter 3 – Methodology. Presents the integrated framework and its platform implementation in detail. The chapter opens with a schematic overview of the pipeline, then describes the data and study area, including the construction and calibration of the panel dataset. The forecasting framework is developed at length: each of the three model families is specified mathematically, feature engineering and training procedures are documented, and the two evaluation protocols (single-window back-testing and 156-fold expanding-window walk-forward cross-validation) are formalised together with their accuracy metrics and the Diebold–Mariano test. The empirical scenario-generation procedure—which bridges the forecasting and optimisation layers—is then described in mathematical detail. The two-stage stochastic optimisation model follows: sets, parameters, decision variables, the objective function, the route-aware recourse penalty, and the constraints are defined explicitly, and the CVaR linearisation that enables linear-programming solution is derived. The chapter closes with the decision-support platform architecture and a tour of its ten tabs through annotated screenshots. **This chapter is intended to be self-contained, providing all the information needed for a qualified reader to replicate or extend the framework.**

Chapter 4 – Computational Study. Specifies the experimental setup in operational detail. The chapter fixes the case-study network and dataset, including the geographic configuration of the five Italian ports, the modal connectivity, and the calibration of the cost, capacity, and emission parameters. It then describes the experimental design for the forecasting evaluation: the training and testing split, the walk-forward fold structure, and the no-tuning policy on hyperparameters. Six representative end-to-end scenarios are defined next, progressively layering forecasting protocol (deterministic baseline, back-testing ML, walk-forward ML), optimisation paradigm (deterministic versus two-stage stochastic, with and without CVaR), and sustainability orientation (cost-focused, balanced, green-focused). The chapter then describes the robustness analyses—a Monte Carlo study across independent scenario seeds and a comparison of alternative scenario-construction methods—and reports the software stack and computational infrastructure used.

Chapter 5 – Results. Reports the empirical findings. The chapter opens with forecasting accuracy across all eight algorithms and ten port–container segments under both evaluation protocols, with statistical significance assessed through the Diebold–Mariano test. The supply–demand balance at the port level follows, together with the optimised repositioning plan under the green-focus configuration as a representative example. The optimisation results are then reported across the six end-to-end scenarios, covering service level, expected cost, CVaR, CO₂ emissions, and modal split, and the Pareto frontier between cost and CO₂ emissions is characterised. The chapter concludes with the Monte Carlo robustness study and a comparison of alternative scenario construction methods.

Chapter 6 – Discussion. Interprets the empirical results against the research questions and hypotheses of the present chapter. The discussion is thematic rather than chronological: it covers why no single forecasting model wins everywhere, how the two-stage recourse channel absorbs forecast-error volume, the bias–variance distinction in what stochastic programming can and cannot fix, the shape of the cost–emission trade-off revealed by the Pareto analysis, and the three layers of uncertainty management (forecasting, recourse, CVaR) provided by the framework. The chapter also draws out the managerial and policy implications for the principal stakeholders in the container-logistics ecosystem, and identifies the boundary conditions within which the findings should be read.

Chapter 7 – Conclusion and Future Work. Summarises the principal findings, revisits the research questions and hypotheses in light of the empirical evidence, positions the contributions against the academic literature and industrial practice, and outlines directions for future research—including multi-period rolling-horizon extensions, multivariate residual modelling that preserves cross-port covariance under joint shocks, and validation on proprietary carrier datasets across larger and more heterogeneous networks.

The next chapter places the contributions of the thesis within the existing literature and articulates the specific research gaps that this work addresses.

Chapter 2

Literature Review

This chapter reviews the academic and industrial literature relevant to the present study and articulates the specific gaps that motivate the methodological contributions developed in subsequent chapters. The management of empty containers sits at the intersection of several research streams: demand forecasting, network optimisation, stochastic programming, and multimodal transport planning. While each of these streams has produced a substantial body of work, they have evolved largely in parallel, with relatively few studies attempting to connect them into a coherent operational framework. The following review traces the development of each stream and identifies the gaps that motivate the present study.

The chapter is organised into six thematic sections. Section 2.1 reviews the empty container logistics literature, focusing on the structural causes of repositioning needs and the early modelling efforts that established the field. Section 2.2 surveys traditional forecasting methods applied in maritime logistics, with particular attention to their strengths and limitations under contemporary operating conditions. Section 2.3 reviews the rapidly growing machine learning literature on container demand forecasting and identifies the studies most directly relevant to empty container management. Section 2.4 examines optimisation models for empty container repositioning, distinguishing deterministic from stochastic formulations and discussing the role of risk measures such as CVaR. Section 2.5 broadens the scope to artificial intelligence in port operations more generally, situating the present work within the emerging research agenda on smart and sustainable ports. Finally, Section 2.6 synthesises the preceding sections, presents a methodological comparison of representative studies, and articulates four specific research gaps that the present thesis is designed to address. The chapter closes with a brief summary in Section 2.7.

2.1 Empty container logistics and the repositioning problem

Empty container repositioning has long been recognised as a structural challenge in containerised transport systems. The root cause is straightforward: global trade flows are inherently asymmetric, meaning that the volume of laden containers entering a port consistently differs from the volume leaving it. Import-dominant ports accumulate surplus empty units that occupy valuable yard space and generate storage costs without producing revenue, while export-oriented ports face chronic equipment shortages that delay shipments and inflate logistics expenses (Song and Carter, 2009; Theofanis and Boile, 2009;

Karmelić et al., 2012). Because empty movements generate no freight revenue yet are operationally indispensable, they represent a major source of inefficiency for shipping lines, terminal operators, and inland logistics providers alike (Stahlbock and Voß, 2010; Braekers et al., 2011).

The coordination of empty container flows is further complicated by the multiplicity of stakeholders involved—shipping lines, depot operators, freight forwarders, inland carriers, and customs authorities—each of whom operates under different incentive structures and planning horizons (Karmelić et al., 2012). External shocks such as port congestion, vessel delays, labour disputes, and geopolitical disruptions can rapidly alter the spatial distribution of empty units in ways that static planning models cannot anticipate (Notteboom et al., 2021; Toygar et al., 2024). The post-pandemic period has provided particularly vivid illustrations of this fragility: container imbalances accumulated during the COVID-19 disruption took years to unwind and exposed the limitations of repositioning systems that had been calibrated to pre-2020 conditions.

Research on the repositioning problem has progressed along two broad tracks. The first focuses on characterising the structure of container imbalances and the logistical mechanisms through which they arise (Song and Carter, 2009; Zain et al., 2014; Theofanis and Boile, 2009). The second develops decision models for determining how surplus containers should be redistributed across the network at minimum cost (Olivo et al., 2005; Shintani et al., 2007; Song and Dong, 2012; Lam, 2014; Abdelshafie et al., 2023). Within this second track, optimisation approaches have ranged from linear and mixed-integer programming formulations for single-period repositioning (Olivo et al., 2005; Epstein et al., 2012) to dynamic programming and Markov decision process models that explicitly account for the sequential nature of inventory management under uncertainty (Sommer et al., 2025; Cai et al., 2023). These studies consistently confirm that coordinated repositioning across an interconnected network produces substantially better outcomes than port-level optimisation in isolation (Song et al., 2022; Abdelshafie et al., 2023).

Despite these advances, the majority of optimisation models in this literature treat demand as a known, exogenous input. Surpluses and shortages are either assumed to be deterministic or estimated from aggregate statistics, rather than derived from data-driven forecasting models calibrated on operational time-series data (Shintani et al., 2007; Lam, 2014; Song et al., 2022; Abdelshafie et al., 2023; Sommer et al., 2025). This treatment limits the operational applicability of existing models, since the very quantity that drives repositioning decisions—future container demand—is inherently uncertain and subject to systematic forecasting error. Bridging this gap between forecasting and optimisation is one of the central methodological objectives of the present thesis, and is addressed in the framework developed in Chapter 3.

2.2 Traditional forecasting methods in maritime logistics

Forecasting has long been used in maritime logistics to support throughput planning, infrastructure expansion, and resource allocation. Classical statistical methods—primarily ARIMA, SARIMA, SARI-MAX, and exponential smoothing variants—remain widely used because they are interpretable, require relatively little data, and can capture regular seasonal patterns in cargo time series (Farhan and Ong, 2018; Lee and Bang, 2024; Dragan et al., 2020). Applications to major international ports have shown

that SARIMA-type models can produce acceptable short-to-medium-term forecasts when cargo series exhibit stable seasonality and limited structural change (Farhan and Ong, 2018; Dragan et al., 2020; Ratmoko and Zuliarso, 2025). The inclusion of exogenous economic indicators such as GDP, trade volumes, and freight indices within SARIMAX frameworks can further improve predictive accuracy during periods of economic expansion (Lee and Bang, 2024; Saeed et al., 2023).

However, classical statistical approaches carry important limitations that constrain their usefulness in the context of empty container repositioning. First, they are inherently linear, making them poorly suited to the nonlinear and often abrupt demand dynamics that characterise modern port operations (Xu et al., 2022; Chan et al., 2019). Second, they are sensitive to structural breaks: a model trained on pre-pandemic data will typically produce large forecast errors when applied to the disrupted conditions of 2020–2022 and the subsequent recovery (Notteboom et al., 2021; Toygar et al., 2024). Third, most statistical models are designed for aggregate throughput and do not distinguish between laden and empty container flows, between different container sizes, or between ports serving different roles in the network (Munim et al., 2023; Dragan et al., 2020). For repositioning purposes, these distinctions are operationally critical: a 20-ft unit and a 40-ft unit are not interchangeable assets, and ports of different functional roles—transshipment hubs versus gateway ports—exhibit distinct demand dynamics (Song and Carter, 2009; Karmelić et al., 2012).

A further limitation is that classical forecasting studies in maritime logistics have rarely been connected to operational decision-making. Their outputs are typically evaluated against standard accuracy metrics—MAE, RMSE, and MAPE—but are not translated into repositioning plans, making it impossible to assess their ultimate impact on operational performance (Chan et al., 2019; Munim et al., 2023; Dragan et al., 2020). This evaluation paradigm, while methodologically convenient, implicitly assumes that improvements in accuracy translate proportionally into improvements in operational outcomes—an assumption that is rarely tested empirically and that, as the present thesis demonstrates in Chapter 5, holds only approximately.

2.3 Machine learning for container demand forecasting

The growing complexity and volatility of maritime demand patterns have stimulated substantial interest in machine learning as an alternative to classical statistical methods (Jahangard et al., 2025; Xu et al., 2022; Kulshrestha et al., 2024). Unlike ARIMA-type models, machine learning approaches can capture nonlinear relationships, heterogeneous feature interactions, and complex temporal dependencies without requiring explicit model specification (Sarker, 2021). Comparative studies conducted across multiple ports and cargo categories have consistently shown that gradient boosting methods (XGBoost, LightGBM, CatBoost), random forests, and deep recurrent architectures outperform classical statistical baselines on container throughput forecasting tasks, often by substantial margins (Xu et al., 2022; Kulshrestha et al., 2024; Ratmoko and Zuliarso, 2025; Taşyürek et al., 2025).

Research has also begun to extend beyond aggregate throughput toward forecasting that is directly relevant to empty container management. Martius et al. (2022) developed a machine learning framework for predicting weekly empty container availability across more than 280 global locations, demonstrating

that neural network and mixture density models substantially outperform conventional approaches on this task. Zugno et al. (2025) proposed a specialised forecasting workflow for empty container repositioning that accounts for the asymmetric cost structure of the problem, showing that forecast-method selection has measurable consequences for repositioning efficiency. Gouaboua et al. (2024) developed forecasting systems for empty container availability in vehicle booking applications, emphasising the operational value of accurate short-term predictions. Diaz et al. (2011) studied empty container volume forecasting at the port level, identifying seasonal patterns and trade structure as key drivers. Henesey and Liu (2020) provided an early comparative analysis of machine learning approaches specifically in the context of empty container repositioning, finding that data-driven methods offer meaningful improvements over naive benchmarks.

A consistent finding across this literature is that forecasting performance is context-dependent: the model that performs best for one port, one container category, or one planning horizon does not necessarily perform best for another (Xu et al., 2022; Chan et al., 2019; Henesey and Liu, 2020). This observation carries direct implications for system design. A framework that applies a single universal forecasting model across all segments of a multi-port network will inevitably sacrifice segment-level predictive performance in segments where a different model would perform better. Yet the majority of existing studies either evaluate a single method per dataset or conduct comparisons that do not translate their findings into differentiated model selection at the segment level (Munim et al., 2023; Dragan et al., 2020; Ratmoko and Zuliarso, 2025). The framework proposed in this thesis explicitly addresses this limitation by selecting the best-performing forecasting model on a per-segment basis, as detailed in Chapter 3.

Despite the progress described above, a fundamental gap remains: forecasting studies almost universally stop at prediction accuracy and do not connect their outputs to operational repositioning decisions (Martius et al., 2022; Zugno et al., 2025; Gouaboua et al., 2024). This creates a structural disconnect between the predictive analytics literature and the optimisation literature reviewed in Section 2.4.

2.4 Optimisation models for empty container repositioning

Optimisation has been central to empty container repositioning research since at least the early 2000s, with formulations ranging from deterministic linear programmes to stochastic and dynamic models (Olivo et al., 2005; Shintani et al., 2007; Lam, 2014; Abdelshafie et al., 2023). Deterministic models typically minimise total repositioning cost subject to supply–demand balance constraints, capacity limits on transport links, and operational feasibility requirements (Olivo et al., 2005; Epstein et al., 2012; Kowshik, 2023). These models provide clear and computationally tractable solutions, but their reliance on point demand estimates means that their recommendations are inherently sensitive to forecast error: a plan optimised for the expected demand may perform poorly when actual demand departs from the prediction (Shintani et al., 2007; Abdelshafie et al., 2023).

Stochastic programming has emerged as the natural extension to address this limitation (Song et al., 2022; Sommer et al., 2025; Cai et al., 2023). In a two-stage stochastic programme, first-stage decisions—such as the quantity of containers to reposition along each route—are made before demand uncertainty

is resolved, while second-stage recourse decisions correct for imbalances that emerge after demand is realised. The expected cost across all scenarios provides the objective to be minimised (Song et al., 2022; Sommer et al., 2025). Risk measures such as Conditional Value-at-Risk (CVaR) can be incorporated into the objective function to control exposure to tail-risk scenarios, making the resulting plans more robust to extreme demand realisations (Song et al., 2022; Vassos et al., 2025). The mathematical formulation of two-stage stochastic programmes with CVaR control, including its linearisation for solution by linear programming, is presented in detail in Chapter 3.

Multi-port extensions of these models have shown that coordinating repositioning decisions across an interconnected network of ports and transport modes produces substantially better outcomes than optimising individual ports in isolation (Song et al., 2022; Abdelshafie et al., 2023; Sadaghi et al., 2024). The inclusion of multiple transport modes—road, rail, and short-sea shipping—introduces genuine trade-offs between cost, transit time, and environmental impact that single-mode models cannot capture (Sharma and Gupta, 2017; Holguín-Veras et al., 2021; Octaviansyah et al., 2024). The persistent weak link in this literature, however, is scenario generation. The majority of multi-port stochastic repositioning models still rely on manually specified or parametrically generated demand scenarios rather than scenarios derived from calibrated forecasting models (Abdelshafie et al., 2023; Sommer et al., 2025). The optimisation then operates on a stylised picture of uncertainty that may bear little resemblance to the errors the forecast actually makes—a coupling problem that motivates the empirical scenario approach used in the present thesis.

A related limitation concerns the treatment of unmet demand in existing stochastic formulations. The standard convention is to apply a single, exogenously specified penalty coefficient to shortage volume, irrespective of the network position of the destination port. In a multi-port network with heterogeneous transport costs across modes and corridors, this uniform-penalty convention can lead to inconsistent valuations of recourse: the optimiser may treat unmet demand at one port as cheaper than routing to it, which is not operationally plausible. The route-aware recourse penalty formulation introduced in Chapter 3 addresses this inconsistency by setting the shortage penalty at each destination port equal to the cheapest reachable emergency repositioning into that port, plus a service-delay surcharge.

2.5 Artificial intelligence in port operations

Beyond container demand forecasting, artificial intelligence has attracted growing interest as a broader enabler of smart port operations and digital maritime logistics (Khazzar et al., 2025; Leng et al., 2025; Jahangard et al., 2025). Bibliometric analyses confirm rapid growth in AI-related port logistics research since 2020, spanning applications in berth allocation, crane scheduling, yard management, vessel arrival forecasting, anomaly detection, and autonomous terminal equipment (Khazzar et al., 2025; Leng et al., 2025). Most of this attention has gone to berth and yard operations rather than to equipment repositioning, leaving the ECR domain comparatively under-served by the broader AI-for-ports research agenda.

Jahangard et al. (2025) conducted a systematic review of 124 papers and identified machine learning and optimisation integration as one of the most promising but underexplored directions in seaport efficiency research, noting that most existing work addresses prediction and prescription separately rather

than jointly. Benghalia et al. (2023) demonstrated how simulation combined with machine learning can improve both efficiency and environmental sustainability in container terminal operations, suggesting that operational and environmental gains can be pursued together rather than as competing objectives. Zheng et al. (2025) applied AI-driven approaches to the container relocation problem in the context of green port management, providing further evidence that AI methods can simultaneously advance operational efficiency and environmental objectives.

These studies paint a consistent picture: AI has demonstrated value across many individual port functions, but integration across the forecasting–optimisation–execution pipeline remains limited (Jahangard et al., 2025; Khazzar et al., 2025; Leng et al., 2025). The field has accumulated a rich toolkit of components but few studies that assemble those components into end-to-end decision-support systems evaluated against realistic operational benchmarks. The decision-support framework and the working platform developed in this thesis represent a step toward closing this gap, integrating forecasting, scenario generation, and stochastic optimisation into a single deployable pipeline.

2.6 Integrated approaches and the research gap

The preceding sections have reviewed each constituent element of the empty container repositioning problem: demand forecasting, stochastic optimisation, multimodal route selection, and risk management. What emerges from this review is that each element has been studied extensively in isolation, but research that integrates them into a unified operational framework—and evaluates the resulting system under realistic conditions—remains scarce.

A number of recent studies have begun to address this integration challenge, although none provides a complete solution for the empty container context. Jahangard et al. (2025) and Zhou et al. (2021) both identify the integration of predictive and prescriptive analytics as a priority research direction in maritime logistics, noting that sequential treatment of forecasting and optimisation as independent sub-problems forfeits the benefits that each could provide to the other. Vassos et al. (2025) proposed an integrated framework for drayage procurement that combines stochastic optimisation with dynamic programming, demonstrating meaningful cost reductions when uncertainty is explicitly modelled; however, their focus is on inland container trucking rather than multi-port maritime rebalancing, and they do not address empty-container repositioning specifically. Cao et al. (2023) combined machine learning forecasts with a branch-and-price optimisation algorithm for liner ship routing, showing that forecast-informed scheduling substantially outperforms schedule-then-forecast approaches; their work is restricted to a single shipping line and does not address empty container repositioning or CVaR risk management. Sadaghi et al. (2024) proposed a simulation-based approach for empty container management at a single port, linking demand prediction to operational decisions, but without a stochastic programming formulation or multi-port scope. Table 2.1 maps the methodological coverage of these and other representative studies against the framework proposed here, making the position of the present thesis within the broader literature visible at a glance.

Table 2.1: Methodological coverage of representative ECR and related logistics frameworks. ✓ = feature included; — = not included or not addressed.

Feature	<i>Olivo et al. (2005)</i>	<i>Shintani et al. (2007)</i>	<i>Lam (2014)</i>	<i>Kuzmierz & Pesch (2019)</i>	<i>Song et al. (2022)</i>	<i>Martius et al. (2022)</i>	<i>Cao et al. (2023)</i>	<i>Abdelshafie et al. (2023)</i>	<i>Liang et al. (2024)</i>	<i>Sommer et al. (2025)</i>	<i>Zugno et al. (2025)</i>	<i>Vassos et al. (2025)</i>	<i>This thesis</i>
ML demand forecasting	—	—	—	—	✓	✓	✓	—	—	—	✓	—	✓
Empirical residual scenarios	—	—	—	—	—	—	—	—	—	—	—	—	✓
Two-stage stochastic programme	—	—	—	—	✓	—	—	✓	✓	—	—	✓	✓
CVaR risk control	—	—	—	—	—	—	—	—	—	—	—	✓	✓
Multimodal transport (road/rail/sea)	—	✓	—	✓	—	—	—	—	—	—	—	—	✓
Cost-CO ₂ trade-off	—	—	—	—	—	—	—	—	—	—	—	—	✓
Multi-port network	✓	✓	✓	✓	✓	✓	✓	✓	✓	—	✓	—	✓
Walk-forward forecast validation	—	—	—	—	—	—	—	—	—	—	✓	—	✓
Segment-specific model selection	—	—	—	—	—	—	—	—	—	—	—	—	✓
Working decision-support platform	✓	—	✓	—	—	—	—	—	—	—	—	—	✓

The pattern in Table 2.1 is informative. Several existing studies provide one or two of the building blocks—multimodal transport, two-stage stochastic programming, machine-learning forecasting, or CVaR risk control—but none combines all of them in a single framework, and none constructs demand scenarios from empirical walk-forward forecast residuals or selects forecasting models on a per-segment basis. The present thesis is the only framework in this comparison that combines all ten features simultaneously.

Taken together, the literature reveals four specific gaps that this thesis is designed to address.

- (i) **No integrated forecast-to-optimisation pipeline for empty container repositioning.** Existing studies treat forecasting and optimisation as separate problems. Forecast outputs are rarely used to construct the demand scenarios that feed stochastic optimisation models, and the operational consequences of forecast error are not systematically traced through the optimisation layer in the empty container context (Abdelshafie et al., 2023; Sommer et al., 2025; Zugno et al., 2025).
- (ii) **Parametric rather than empirical scenario generation.** When stochastic models do incorporate uncertainty, scenarios are typically drawn from parametric distributions (most commonly a normal distribution fitted to aggregate statistics) rather than from the empirical residuals of a calibrated forecasting model. This decouples the optimisation model from the actual error characteristics of the forecast and leads to miscalibrated CVaR estimates (Song et al., 2022; Vassos et al., 2025).
- (iii) **Absence of port- and container-type-specific model selection.** Most forecasting studies apply a single model across all segments of a multi-port network, despite consistent empirical evidence that the best-performing model varies across ports, container categories, and planning horizons (Xu et al., 2022; Chan et al., 2019; Henesey and Liu, 2020). This limits predictive accuracy and, consequently, the quality of the demand scenarios fed into the optimisation layer.

- (iv) **Limited multimodal and multi-objective scope.** Many repositioning models consider only a single transport mode or a single objective. Real-world repositioning decisions involve genuine trade-offs between cost, transit time, and CO₂ emissions across road, rail, and maritime modes, and sustainable logistics practice increasingly demands that these trade-offs be made explicit (Sharma and Gupta, 2017; Holguín-Veras et al., 2021; Octaviansyah et al., 2024).

2.7 Chapter summary

This chapter has reviewed the academic and industrial literature relevant to the empty container repositioning problem across six thematic areas. The review has shown that each constituent element—demand forecasting, optimisation under uncertainty, multimodal network planning, and risk-aware decision-making—has been studied extensively, producing a rich and methodologically diverse body of work.

At the same time, the literature reveals a consistent and important fragmentation: forecasting studies and optimisation studies have developed largely in parallel, with little systematic effort to connect predictive accuracy to prescriptive value. Where integration has been attempted, it has typically been confined to specific sub-problems (drayage, liner routing, single-port management) rather than to the multi-port empty container repositioning problem in its full operational complexity. The methodological comparison in Table 2.1 makes this fragmentation explicit: no existing study covers the full set of features that the present thesis brings together.

The four gaps identified in Section 2.6—the absence of an integrated forecast-to-optimisation pipeline, the prevalence of parametric rather than empirical scenario generation, the lack of port- and container-type-specific model selection, and the limited multimodal and multi-objective scope—together motivate the methodological framework developed in Chapter 3. The chapter that follows responds to each of these gaps in turn, designing an integrated decision-support framework that combines port- and container-type-specific machine learning forecasting with empirical scenario generation, two-stage stochastic optimisation under CVaR, and multimodal transport planning across road, rail, and short-sea shipping modes, and operationalises it as a working decision-support platform.

Chapter 3

Methodology

This chapter presents the integrated framework developed in this thesis and its operationalisation as a working decision-support platform. The chapter is structured to be self-contained: every methodological component is specified in mathematical detail, every parameter is defined explicitly, and every design choice is justified against the gaps identified in Chapter 2. A qualified reader should be able to replicate or extend the framework on the basis of the material in this chapter alone.

The chapter is organised as follows. Section 3.1 provides a schematic overview of the complete pipeline and introduces the two-step structure that links forecasting and optimisation. Section 3.2 describes the data and study area, including the construction and calibration of the panel dataset that supports the empirical analysis. Section 3.3 develops the forecasting framework in depth: the eight-algorithm portfolio organised into three families, the two complementary evaluation protocols (back-testing and walk-forward), the accuracy metrics, the Diebold–Mariano test, and the segment-specific model selection rule. Section 3.4 formalises how the selected forecasts produce port-level demand estimates and how their walk-forward residuals are sampled into the demand scenarios that feed the optimiser; Sections 3.5 and 3.6 introduce the stock initialisation and the supply–demand balance that the optimiser uses to identify surplus and deficit nodes. Section 3.7 parameterises the multimodal transport network across road, rail, and short-sea shipping. Section 3.8 formulates the two-stage stochastic optimisation model with CVaR control, including the route-aware recourse penalty that is one of the methodological contributions of this thesis. Finally, Section 3.9 documents the decision-support platform that operationalises the framework as software, illustrated through annotated screenshots of its ten-tab interface, and Section 3.10 provides a short chapter summary.

3.1 Framework overview

This thesis proposes an integrated decision-support framework for empty container repositioning management, organised around two operational steps—forecasting and optimisation—and delivered as a working decision-support platform. The forecasting step produces segment-level demand forecasts from historical data using machine learning, and the optimisation step uses those forecasts to generate a repositioning plan across the multimodal network. Figure 3.1 shows the full data flow; the two steps and the

supporting components are detailed in the subsections that follow.

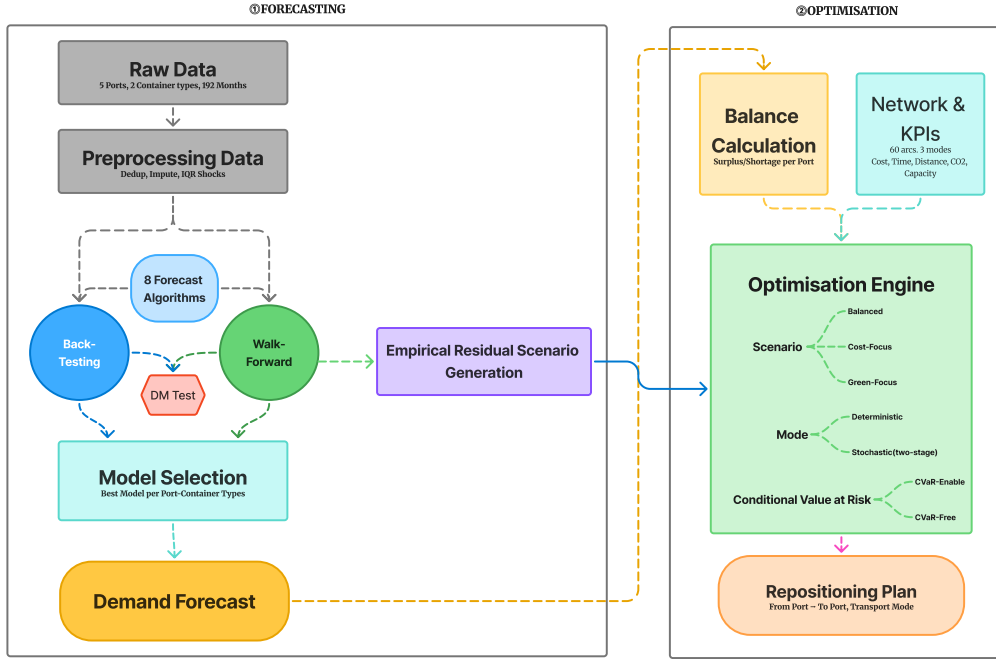


Figure 3.1: Schematic overview of the integrated forecast-driven stochastic optimisation framework. Historical data feed an eight-algorithm forecasting portfolio whose walk-forward residuals are sampled to construct the empirical scenario set that drives the two-stage stochastic optimisation model with CVaR control.

Three design principles distinguish this framework from prior work in the empty-container literature. The first is the explicit coupling between the predictive and prescriptive layers. Rather than treating forecasting and optimisation as independent sub-problems, the framework threads the output of the forecasting layer—specifically, the empirical residual distribution of the selected model—into the scenario set that drives the optimisation layer. The forecaster and the optimiser therefore speak the same statistical language: the optimisation model responds to the errors that the forecasting model actually makes, rather than to a parametric proxy for them.

The second principle is segment-specific model selection. The forecasting portfolio comprises eight algorithms organised into three families, and each algorithm is trained and evaluated independently for every port–container-type segment. The framework selects the best-performing algorithm separately for each segment rather than imposing a single model across the entire network, in recognition of the empirical regularity that no single forecasting method dominates across heterogeneous demand structures (Xu et al., 2022; Chan et al., 2019; Henesey and Liu, 2020).

The third principle is dual-protocol evaluation. Each algorithm is evaluated under two complementary protocols: a single-window back-testing benchmark and a 156-fold expanding-window walk-forward cross-validation. The back-testing protocol provides a familiar single-split diagnostic, while the walk-forward protocol mirrors the information state under live deployment by retraining the model at

every planning cycle. The two protocols may select different best models for the same segment, and the framework retains both selections in parallel rather than collapsing them into a single recommendation. This dual-branch architecture surfaces the operationally honest figure at every step of the analysis instead of burying it in a single summary statistic.

The framework is operationalised as a working decision-support platform that exposes each pipeline stage as an independently auditable module through a tabbed interface. The platform’s ten tabs partition the workflow into four logical phases: data ingest (Tab 1), forecast and evaluation (Tabs 2–4), optimisation set-up (Tabs 5–7), and optimisation, results, and diagnostics (Tabs 8–12). Section 3.9 describes the platform architecture in detail.

3.2 Data and study area

3.2.1 Port network and coverage

The empirical study covers a network of five major Italian seaports that together handle more than 60% of national containerised trade. The five ports span structurally different operational roles within the Mediterranean container system. **Genova** and **Gioia Tauro** function as chronic surplus nodes: Genova is the principal gateway port for Northern Italy and Switzerland, while Gioia Tauro is one of the largest transshipment hubs in the Mediterranean. **La Spezia**, **Livorno**, and **Trieste** operate as persistent deficit nodes, with La Spezia and Livorno serving the Tyrrhenian export corridors and Trieste serving the Central and Eastern European hinterland through its Adriatic position. This combination of surplus and deficit roles mirrors the long-standing imbalances documented in Italian port statistics (Assoporti, 2023) and provides a realistic testbed for empty container repositioning across a multi-port multimodal network.

3.2.2 Panel structure and dataset

The dataset is organised as a port–container-type monthly panel covering 192 monthly observations per series from January 2010 through December 2025. Two container types are modelled in parallel: standard 20-foot dry van units (20DV) and 40-foot high-cube units (40HC), reflecting the dominant operational distinction in European container traffic. The full panel therefore comprises

$$5 \text{ ports} \times 2 \text{ container types} \times 192 \text{ months} = 1,920 \text{ observations.} \quad (3.1)$$

For each port–container–month combination the dataset records four directional flows: empty outbound (`Empty_Out_TEU`), empty inbound (`Empty_In_TEU`), full outbound (`Full_Out_TEU`), and full inbound (`Full_In_TEU`). All quantities are expressed in TEU. The forecasting layer treats the empty outbound flow as the target variable: this is the operationally relevant quantity for the repositioning planner, since it corresponds to the demand for empty equipment that the port must satisfy from its own stock or from inbound repositioning shipments. The remaining three flows are retained as candidate explanatory features and are available to the gradient-boosting models that admit multivariate input vectors.

3.2.3 Temporal coverage and operating regimes

The 16-year window deliberately spans three structurally distinct demand regimes. The first regime, from 2010 to 2019, provides a long pre-disruption baseline characterised by stable seasonal patterns and gradual trend growth. The second regime, from 2020 to 2022, captures the COVID-19 demand shock, the subsequent container shortage crisis, and the freight-rate volatility that followed; this regime contains the most demanding structural break in the panel. The third regime, from 2023 to 2025, covers the post-pandemic period of geopolitical freight volatility—Suez and Red Sea re-routing, supply-chain re-configuration, and the incipient effects of European maritime decarbonisation policy. Retaining all three regimes in the training and evaluation windows exposes the forecasting models to a more demanding out-of-sample test than a single-regime dataset would, and tests the optimisation layer against the kind of demand uncertainty that motivates the framework in the first place (Notteboom et al., 2021; Toygar et al., 2024; United Nations Conference on Trade and Development, 2023).

3.2.4 Data construction and calibration boundary

Disaggregated port–container–month flow data are not publicly released by carriers or terminal operators (Song and Carter, 2009; Vassos et al., 2025; Sommer et al., 2025). Following established practice in stochastic logistics research when carrier-level records are inaccessible, this thesis therefore works with a *calibrated* monthly series rather than raw operational records. The contribution of such studies rests on the modelling framework and the structural insights it produces rather than on the confidentiality of the underlying figures.

The calibration procedure is anchored to four public sources. Port-level annual volumes are taken from Assoporti and used to set the level of each series. Within-year seasonal profiles are extracted by seasonal–trend decomposition (STL) of Eurostat monthly country-level data and projected onto each port proportionally to its national-share weight. Inter-annual variability is modelled with a coefficient of variation of 0.08, calibrated against the empirical volatility of the Eurostat aggregate over the same window. The 20DV/40HC composition is set to 42/58 in line with the European average reported by Eurostat for containerised trade. The 2020–2021 COVID-19 contraction is preserved explicitly at all five ports, so that the subsequent validation exercises face genuine structural breaks rather than smooth-demand regimes only (Notteboom et al., 2021; United Nations Conference on Trade and Development, 2023).

The resulting dataset is not a substitute for proprietary carrier data; it is a controlled, reproducible approximation that preserves the structural imbalances, seasonality, and disruption signatures of the underlying flows. **Absolute TEU volumes, monetary costs, and emission figures should therefore be interpreted within this calibration boundary.** The methodological contributions of the thesis and the relative comparisons across configurations—which constitute the core of the empirical results—do not depend on the precise absolute values.

3.2.5 Quality pipeline

A structured data-quality pipeline runs before any modelling. Duplicate observations are resolved under an upsert policy that retains the most recent revision. Missing months are forward-filled and flagged with an `is_imputed` indicator, so that downstream diagnostics can distinguish imputed values from observed ones. Outliers are tagged but not removed: an interquartile-range criterion ($\pm 3 \times \text{IQR}$) flags extreme values as anomalies, but they are retained in the series with an `is_outlier` marker rather than imputed away. This choice is deliberate. The 2020 COVID-19 contraction would itself be flagged as an outlier under a naive cleaning rule, and removing it would defeat the purpose of including the disruption window in the panel. The stochastic optimisation layer is designed to hedge against precisely such events; preserving them in the training data is essential to that hedge being informed by real disruption signatures rather than by a smoothed-out demand history.

3.2.6 Data ingest in the decision-support platform

The data ingest step is operationalised in the first tab of the decision-support platform (Figure 3.2). The user selects the calibrated CSV file from the file picker, and the platform performs three automatic actions on load: it counts the rows, identifies the date range, and enumerates the ports and container types detected in the file. The interface then displays a paginated preview of the raw flows so that the user can verify that the file structure matches the expected schema before any modelling step is invoked. This first tab is intentionally minimal: it surfaces only the information needed to confirm that the dataset has been loaded correctly, and defers all modelling decisions to the downstream tabs documented in the following sections.

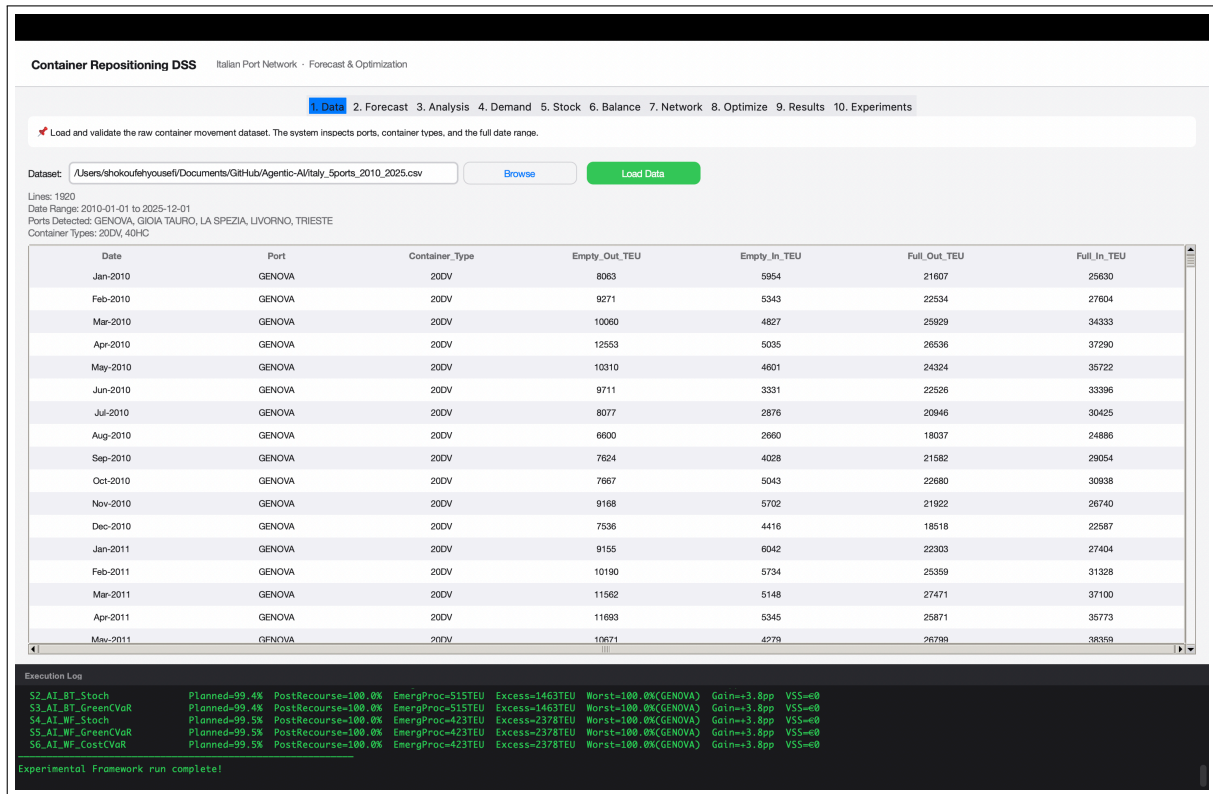


Figure 3.2: Tab 1 – *Data*. Calibrated panel ingest showing detected ports (Genova, Gioia Tauro, La Spezia, Livorno, Trieste), container types (20DV, 40HC), date range (January 2010 to December 2025), and a row-level preview of the empty and full inbound/outbound flows used downstream.

3.3 Forecasting framework

The forecasting framework converts the calibrated panel of Section 3.2 into segment-level demand forecasts that feed the downstream optimisation layer. The framework is organised around three design decisions: a portfolio of eight algorithms grouped into three families (Section 3.3.1), two complementary out-of-sample evaluation protocols (Section 3.3.2), and a segment-specific selection rule that picks the best algorithm separately for each port–container-type combination (Section 3.3.6). The pairwise Diebold–Mariano test (Section 3.3.5) verifies that the differences underlying the selection rule are statistically significant rather than artefacts of sampling.

3.3.1 Forecasting model portfolio

The portfolio comprises eight algorithms organised into three families: three classical baselines, four tree-based machine learning models, and one inverse-MAE weighted ensemble combining the seven individual learners. Each algorithm is fitted independently for every port–container segment, with no cross-segment information sharing during training. This per-segment training policy is deliberate: it isolates the demand dynamics of each segment and allows the segment-specific selection rule of Section 3.3.6 to pick a different best algorithm for each port–container combination without fitting confounds.

Baseline models

Three classical baselines set the performance floor against which the more flexible methods are evaluated (Chan et al., 2019). They share two properties: they are parameter-free or near parameter-free, and they are commonly used in operational practice. A new method that fails to beat all three baselines decisively offers no practical value.

Naïve 3-Month (Naïve_3M). The forecast at time $t + 1$ is the arithmetic mean of the three most recent observations:

$$\hat{y}_{t+1}^{\text{N3M}} = \frac{1}{3} \sum_{k=0}^2 y_{t-k}. \quad (3.2)$$

Drift. The Drift method extrapolates the average period-over-period change observed in the training window:

$$\hat{y}_{t+1}^{\text{Drift}} = y_t + \frac{1}{T-1} (y_t - y_1), \quad (3.3)$$

where T is the length of the training window. The method captures a constant linear trend but assumes that the trend continues unchanged into the forecast horizon.

Seasonal Naïve. The Seasonal Naïve forecast reuses the observation from the same month of the previous year:

$$\hat{y}_{t+1}^{\text{SN}} = y_{t+1-12}, \quad (3.4)$$

which preserves the annual seasonal pattern but ignores trend and any other non-seasonal structure.

Machine learning models

Four tree-based machine learning models complete the individual-model set: Random Forest, XGBoost, LightGBM, and CatBoost. Random Forest is included as a bagging-based reference (Breiman, 2001); the remaining three are gradient-boosting variants that differ in their splitting and regularisation strategies (Chen and Guestrin, 2016; Ke et al., 2017; Prokhorenkova et al., 2018). Their inclusion as a group, rather than a single representative model, is motivated by the well-documented finding that no single gradient-boosting variant dominates uniformly across datasets (Sarker, 2021; Xu et al., 2022); including all four exposes the segment-level selection rule to a non-degenerate set of competitive learners.

All four models are trained on a common feature vector constructed from the target series itself, with no exogenous covariates. The feature vector comprises:

- *Temporal lags* at horizons $\ell \in \{1, 2, 3, 6, 12\}$ months, capturing short-term, quarterly, and annual dependencies;
- *Rolling means* over the previous 3 and 6 months, capturing local trend;
- *Calendar indicators*: month-of-year, quarter, and year, capturing seasonal and structural effects.

The deliberate restriction to lag-based features preserves the comparability of the machine learning models with the three classical baselines, which are also univariate. Hyperparameters are held at library defaults across all four models; the framework deliberately does not perform per-segment hyperparameter tuning, which would inflate the gap between training and live performance and undermine the realism of the walk-forward evaluation.

Weighted ensemble

The ensemble combines the seven individual models—three baselines and four machine-learning learners—into a single predictor through inverse-MAE weighting:

$$\hat{y}_t^{\text{ens}} = \sum_{m=1}^M w_m \hat{y}_{m,t}, \quad w_m = \frac{1/\text{MAE}_m^{\text{prior}}}{\sum_{m'=1}^M 1/\text{MAE}_{m'}^{\text{prior}}}, \quad \sum_{m=1}^M w_m = 1, \quad M = 7, \quad (3.5)$$

where $\text{MAE}_m^{\text{prior}}$ is the mean absolute error of model m measured on prior folds of the walk-forward protocol described in Section 3.3.2. The weighting therefore relies only on past out-of-sample performance, which keeps the combination free of look-ahead bias: at every fold k , the weights used to combine the seven base models are computed from the errors observed on folds $0, \dots, k-1$, and never use information from fold k itself.

Two properties of this construction are worth noting. First, the weighting penalises high-error models proportionally to their error: a model whose MAE is twice that of another receives half the weight. Second, the ensemble is treated as a separate eighth algorithm in the selection rule of Section 3.3.6 rather than as a default choice; in segments where a single individual learner performs substantially better than all others, the selection rule may prefer that learner over the ensemble.

3.3.2 Evaluation protocols

Each algorithm is evaluated under two complementary out-of-sample protocols: a single-window *back-testing* benchmark and an expanding-window *walk-forward* cross-validation. The two protocols test different aspects of model behaviour, and the framework retains both rather than collapsing them into a single recommendation.

Back-testing

The back-testing protocol splits the full panel once at a calendar boundary: the training window covers January 2010 through December 2024 (180 months), and the test window covers January 2025 through December 2025 (12 months). Each algorithm is fitted once on the training window and produces a single trajectory of one-step-ahead forecasts across the test window, against which the standard accuracy metrics of Section 3.3.4 are computed.

Back-testing offers a simple single-window diagnostic that is intuitive for practitioners and useful for stable series. Its weakness is that it conditions on a single test window, so a model that happened to fit the particular 12-month period well may receive an inflated performance estimate that does not generalise.

This is particularly acute when the test window falls outside the disruption regime of 2020–2022: a model evaluated only on the 2025 calendar window will not be tested against the most demanding structural breaks in the panel.

Walk-forward cross-validation

The walk-forward protocol addresses this weakness directly by evaluating each algorithm under conditions that mirror live deployment, where every forecast must rely on information available up to its own origin. The protocol uses an expanding training window with $K = 156$ folds. Fold $k \in \{0, 1, \dots, K - 1\}$ trains the algorithm on the first $\tau_{\min} + k$ observations, where $\tau_{\min} = 36$ months is the minimum training window, and produces a one-step-ahead forecast for month $\tau_{\min} + k + 1$. The training window therefore expands by one month at every fold; this provides the longest possible training history at each origin while preserving strict separation between training and test data. The full schedule is illustrated in Figure 3.3.

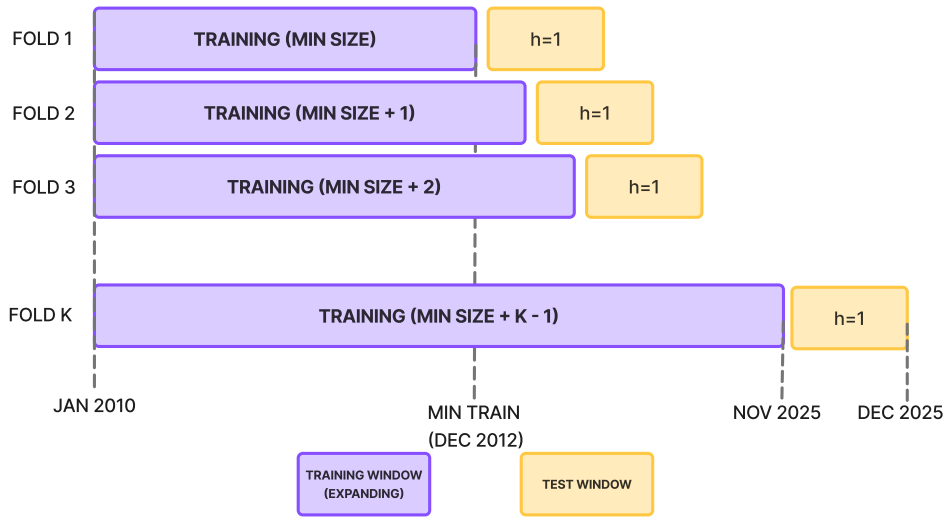


Figure 3.3: Walk-forward cross-validation protocol with $K = 156$ expanding folds. Each fold adds one observation to the training window (minimum 36 months) and produces a one-step-ahead forecast at horizon $h = 1$. Each model is refitted from scratch at every fold, so its parameters never see data beyond the forecast origin.

Three properties of this protocol are essential to its operational realism. First, each algorithm is refitted from scratch at every fold rather than warm-started from the previous fold; this ensures that the parameters at fold k never depend on data beyond month $\tau_{\min} + k$, even indirectly. Second, fold residuals accumulate across the walk-forward sequence: the residual set available at the end of fold $k - 1$ contains k out-of-sample errors per algorithm and segment, and this set is the input to the ensemble weighting rule (3.5) at fold k . The estimates therefore become progressively more precise as the walk-forward

window expands. Third, the protocol delivers a long sequence of one-step-ahead forecasts that spans the full disruption regime of 2020–2022, which back-testing on the 2025 window cannot.

The walk-forward protocol is the basis used for the rest of the pipeline. The empirical scenarios that drive the optimisation layer are sampled from the walk-forward residuals (Section 3.4.2), and the operationally honest demand forecast carried into the optimiser is the walk-forward forecast for the December target month. The back-testing protocol is retained alongside it as a diagnostic that exposes how much of the apparent accuracy advantage of any single window is robust to the more demanding evaluation.

3.3.3 Forecasting in the decision-support platform

The forecasting framework is operationalised in the second tab of the decision-support platform, which exposes the two evaluation protocols and the pairwise Diebold–Mariano test through three sub-views. The *Backtesting* sub-view (Figure 3.4) reports the single-window MAE, RMSE, and MAPE for each of the eight algorithms across the ten port–container segments, together with the per-segment best model highlighted in the lower panel. The *Walk-Forward* sub-view (Figure 3.5) reports the analogous metrics under the expanding-window protocol and adds a bar chart that visualises per-segment performance across all eight algorithms, making the algorithm-by-segment heterogeneity immediately apparent. The *DM Test* sub-view (Figure 3.6) executes the pairwise Diebold–Mariano comparisons formalised in Section 3.3.5 and tabulates the DM statistic, the p -value, and the significance flag at the 3% level for every model pair within every segment. The three sub-views together expose the forecasting layer as an auditable component of the pipeline rather than as a black box that produces a single demand number.

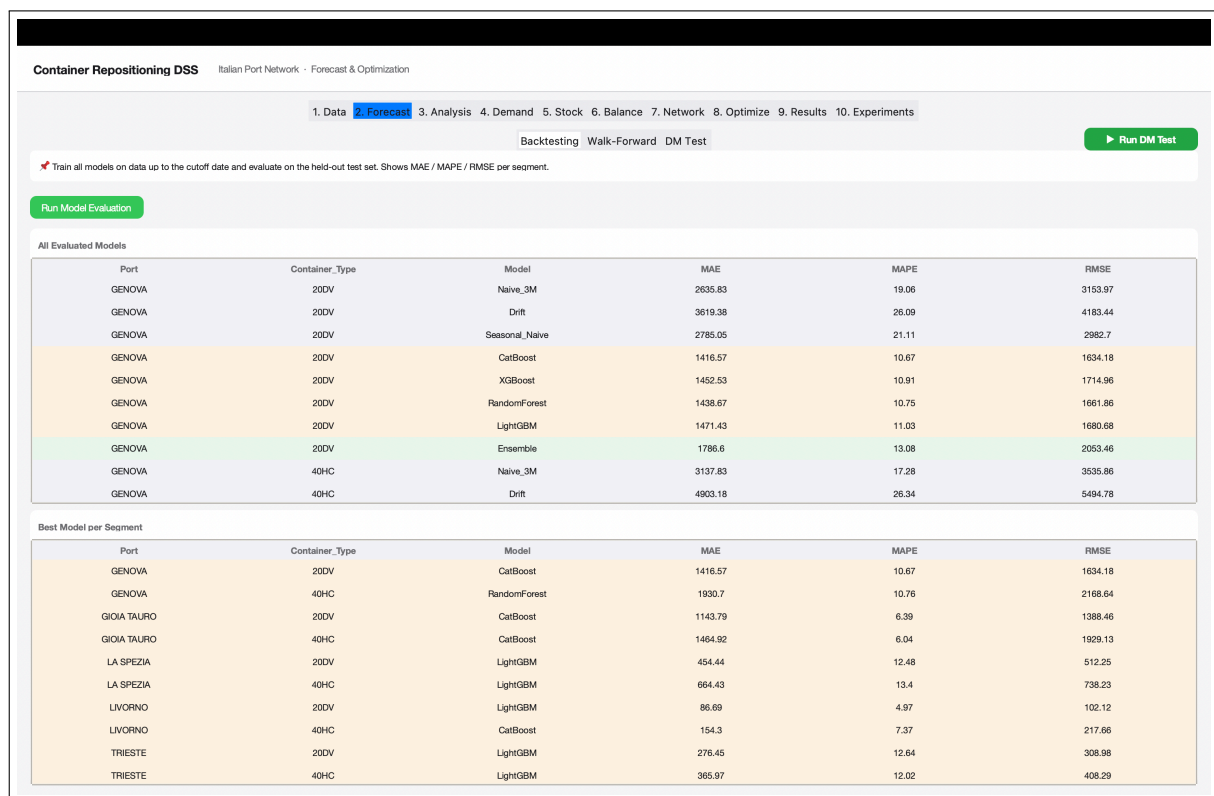


Figure 3.4: Tab 2 – *Forecast, Backtesting* sub-view. Single-window evaluation (January 2010–December 2024 training, calendar 2025 testing) of all eight algorithms across the ten port–container segments, with the per-segment best-model panel highlighted in the lower table.

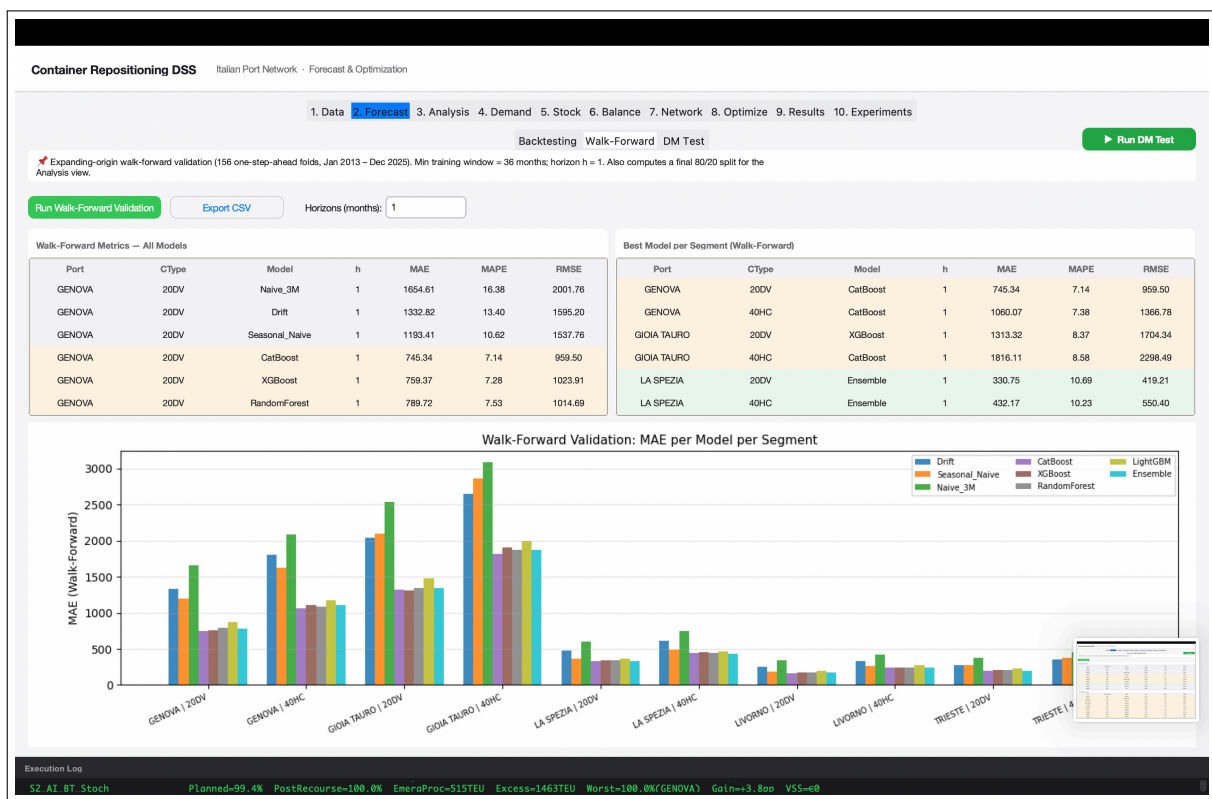


Figure 3.5: Tab 2 – *Forecast*, *Walk-Forward* sub-view. Expanding-window cross-validation with $K = 156$ folds, showing the all-models table on the upper left, the per-segment best-model table on the upper right, and a bar chart of MAE per model per segment in the lower panel that exposes the algorithm-by-segment heterogeneity at a glance.

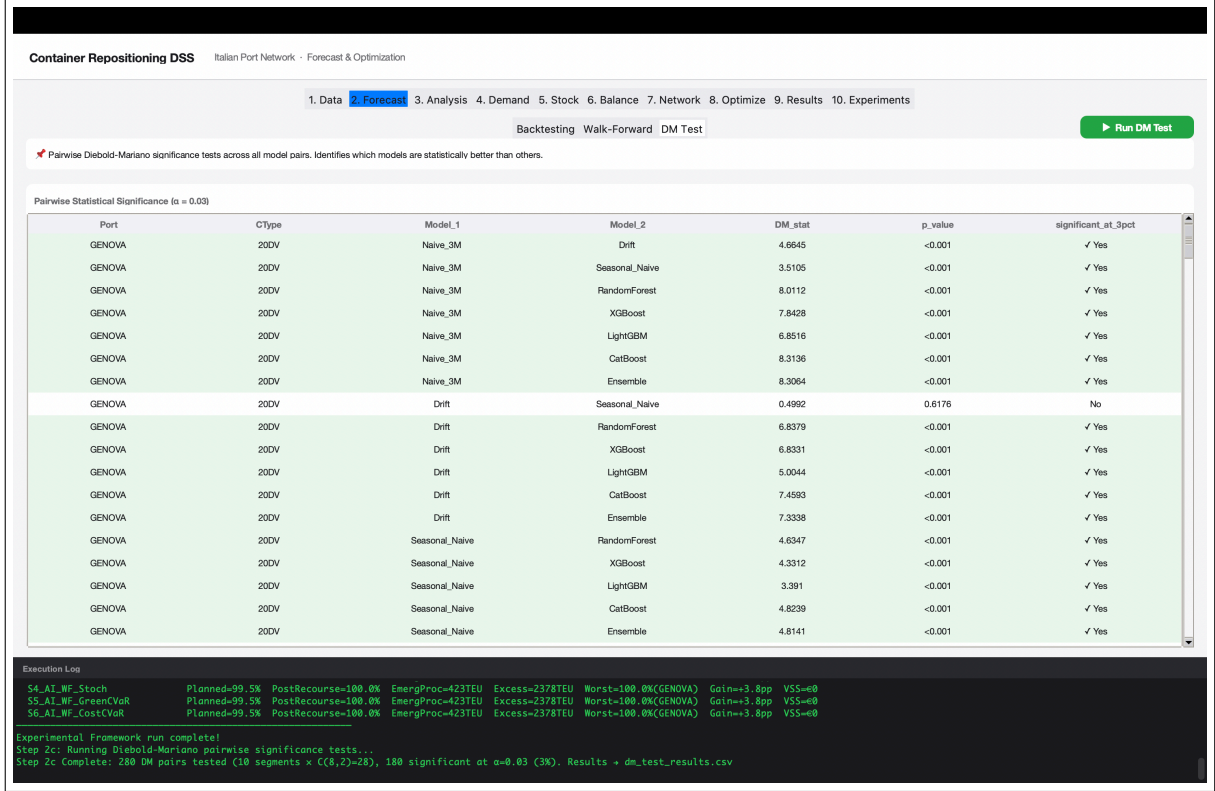


Figure 3.6: Tab 2 – *Forecast*, *DM Test* sub-view. Pairwise Diebold–Mariano significance tests across all $\binom{8}{2} = 28$ model pairs within each port–container segment, with the DM statistic, the p -value, and the significance flag at the 3% level reported per pair.

3.3.4 Accuracy metrics

Three standard accuracy metrics are computed for every algorithm–segment–protocol triple: mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). The three are retained in parallel because they capture different aspects of forecast error and are not redundant (Hyndman and Koehler, 2006; Chan et al., 2019).

Let $\{y_t\}_{t=1}^N$ denote the realised target values and $\{\hat{y}_t\}_{t=1}^N$ the corresponding out-of-sample forecasts, where N is the number of test points (one trajectory under back-testing, the walk-forward fold count under walk-forward). The three metrics are defined as

$$\text{MAE} = \frac{1}{N} \sum_{t=1}^N |y_t - \hat{y}_t|, \quad (3.6)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \hat{y}_t)^2}, \quad (3.7)$$

$$\text{MAPE} = \frac{100}{N} \sum_{t=1}^N \left| \frac{y_t - \hat{y}_t}{y_t} \right|. \quad (3.8)$$

The three metrics serve distinct diagnostic roles. MAE summarises the average error magnitude

on the original TEU scale and is the natural quantity for an operational planner who has to absorb the typical miss; RMSE penalises larger deviations more heavily through the quadratic loss and is therefore sensitive to occasional sharp misses that a service-level commitment cannot tolerate; MAPE expresses the error as a proportion of demand and allows cross-segment comparison between ports of very different size. Lower values are preferred on all three. MAE is the primary selection metric used by the rule in Section 3.3.6; RMSE and MAPE are retained as diagnostic checks that flag segments where the three metrics disagree on the best algorithm.

3.3.5 Pairwise model comparison: Diebold–Mariano test

The accuracy ranking produced by Section 3.3.4 is a point estimate. Two algorithms that produce different MAE values on the same segment may or may not be *statistically* distinguishable: the observed difference could reflect a real superiority of one algorithm over the other, or it could be an artefact of the particular sequence of forecasts that the test window produced. The Diebold–Mariano (DM) test (Diebold and Mariano, 1995; Harvey et al., 1997) formalises this distinction by testing the null hypothesis that two algorithms have equal expected predictive accuracy against the alternative that they do not.

For a given segment, let $e_{m,t} = y_t - \hat{y}_{m,t}$ denote the forecast error of algorithm m at time t , and let $\ell(\cdot)$ be a loss function. The DM test considers the loss-differential series

$$d_t = \ell(e_{m_1,t}) - \ell(e_{m_2,t}), \quad t = 1, \dots, T, \quad (3.9)$$

between two competing algorithms m_1 and m_2 . The test statistic is

$$\text{DM} = \frac{\bar{d}}{\sqrt{\hat{V}(\bar{d})/T}}, \quad \bar{d} = \frac{1}{T} \sum_{t=1}^T d_t, \quad (3.10)$$

where $\hat{V}(\bar{d})$ is the Newey–West HAC (heteroskedasticity- and autocorrelation-consistent) estimator of the long-run variance of $\bar{d}$, with bandwidth equal to the forecast horizon (Newey and West, 1987). Under the null hypothesis of equal predictive accuracy, the test statistic is asymptotically standard normal.

The implementation in this thesis uses squared-error loss $\ell(e) = e^2$, which makes the test consistent with RMSE as the underlying scoring rule and ensures that large misses receive the same weight in the significance test as they do in the metric. For each port–container segment, all $\binom{8}{2} = 28$ pairwise comparisons across the eight-algorithm portfolio are computed, yielding $10 \times 28 = 280$ tests in total. The significance threshold is set at $\alpha = 0.03$ rather than the conventional $\alpha = 0.05$. The tighter cut-off provides a partial control of family-wise error across the 28 simultaneous comparisons within each segment without resorting to a full Bonferroni correction, which would be overly conservative given the strong overlap between the eight algorithms.

The empirical results of these 280 tests are reported in Chapter 5. The test does not by itself nominate a winning algorithm; it merely partitions the pairs of algorithms within each segment into those that are statistically distinguishable and those that are not. The selection rule of the next subsection draws on this information to qualify, rather than to replace, the MAE-based ranking.

3.3.6 Best-model selection rule

Given the metrics of Section 3.3.4 and the DM tests of Section 3.3.5, a single algorithm is selected per port–container segment. The selection rule is the lowest-MAE criterion:

$$m_{j,c}^* = \arg \min_{m \in \mathcal{M}} \text{MAE}_{m,j,c}, \quad (3.11)$$

where $\mathcal{M}$ is the eight-algorithm portfolio of Section 3.3.1, j indexes the five ports, and c indexes the two container types. The selection is carried out independently for the back-testing and the walk-forward protocols, yielding two parallel best-model sets:

$$m_{j,c}^{*,\text{BT}}, \quad m_{j,c}^{*,\text{WF}}, \quad j \in \mathcal{P}, \quad c \in \{20\text{DV}, 40\text{HC}\}. \quad (3.12)$$

These two sets may select different algorithms in segments where the two protocols disagree, and the framework deliberately retains both rather than collapsing them into a single recommendation. The walk-forward selection drives the demand forecast that is carried into the optimisation layer (Section 3.4); the back-testing selection is retained alongside it as a diagnostic baseline that exposes how much of the apparent accuracy advantage of any single window is robust to the more demanding evaluation.

Two safeguards are applied in parallel with the selection rule. First, RMSE and MAPE are inspected as diagnostic checks: when the three metrics disagree on the best algorithm, the corresponding segment is flagged for manual review rather than being silently overridden. Second, the Diebold–Mariano tests verify that the MAE gap between the selected algorithm and its closest competitor is statistically significant. When the gap is not significant at the $\alpha = 0.03$ level, both algorithms are reported as a tied selection, and the operationally chosen algorithm is flagged accordingly. The two safeguards are deliberately conservative: they do not change the algorithm chosen by Eq. (3.11), but they surface the segments where that choice is fragile so that downstream diagnostics can address the fragility explicitly rather than inherit it silently.

3.3.7 Forecast analysis in the decision-support platform

The forecast analysis step is operationalised in the third tab of the decision-support platform, which exposes the per-segment forecast trajectories under both evaluation protocols and provides the user with the explicit choice of which protocol to carry into the downstream demand step. Three views of the tab are shown in Figures 3.7, 3.8, and 3.9.

The user selects a segment from the dropdown (GENOVA | 20DV in the figure) and the platform displays two summary blocks side by side: the *Backtesting Best* block on the left, which reports the BT best-model identity and the corresponding single-window MAE, MAPE, and RMSE; and the *Walk-Forward Best* block on the right, which reports the WF best-model identity and the analogous walk-forward metrics. The two blocks make the dual-protocol selection of Eq. (3.12) visible at a glance, and the comparison frequently exposes substantial differences—in the figure, the WF MAE of 745.34 TEU is nearly half the BT MAE of 1,416.57 TEU on the same segment, even though both protocols select

CatBoost.

The lower panel of the tab plots the realised series against the one-step-ahead forecasts of all eight algorithms over the test window. The *Backtesting Forecast Comparison* view (Figure 3.7) shows the trajectories under the back-testing protocol on the calendar 2025 test window; the *Walk-Forward Forecast Comparison* view (Figure 3.8) shows the analogous trajectories under the expanding-window walk-forward protocol. The two views expose how much the apparent goodness-of-fit of a single window can mislead: under walk-forward the same forecasters track the realised series more tightly, but they also disagree more visibly with each other during the disrupted months, which is the operationally relevant signal.

The *Show 90% Prediction Intervals* checkbox overlays a 90% prediction band on the selected algorithm's forecast (Figure 3.9). The intervals are constructed from the empirical walk-forward residual distribution of Section 3.4.2 by taking the 5th and 95th percentiles of the residuals; the coverage rate of the realised values inside the band is displayed in the top-right corner of the chart and provides an immediate calibration diagnostic. Where the coverage rate deviates substantially from the nominal 90%, the prediction band is mis-calibrated and the analyst is warned that the underlying residual distribution may not be representative of the deployment window.

The right-hand *Use for Demand* radio buttons formalise the protocol choice. Selecting BT carries the back-testing forecast into the demand step, while selecting WF carries the walk-forward forecast. The default selection is WF, in line with the framework's design principle that the operationally honest figure is the one carried into downstream optimisation. The *Accept Forecast* button commits the choice and unlocks the demand tab; the *Edit Manually* button allows the analyst to override individual forecasts before they are passed on, which is useful when external information not present in the calibrated dataset (a known seasonal disruption, a scheduled vessel call, a contract change) should be reflected in the planning input.

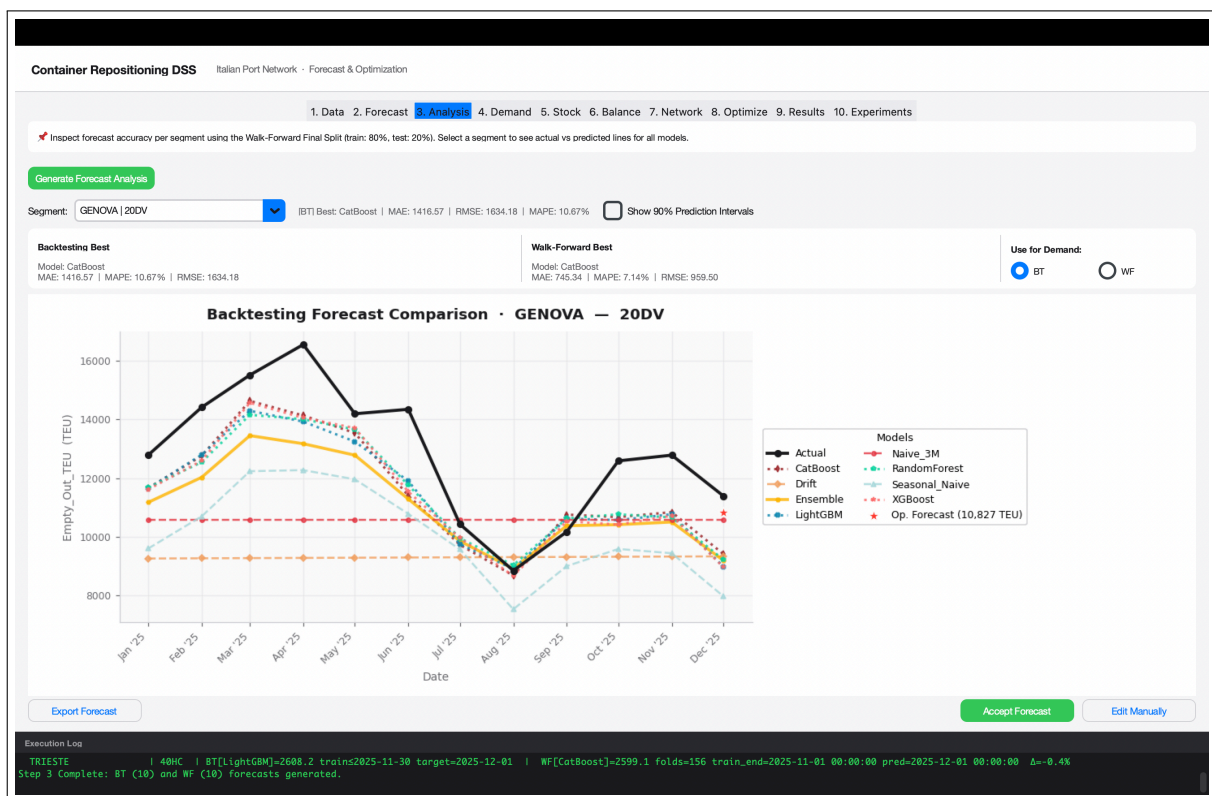


Figure 3.7: Tab 3 – *Analysis*, back-testing view. Realised series against the one-step-ahead forecasts of all eight algorithms over the calendar 2025 test window.

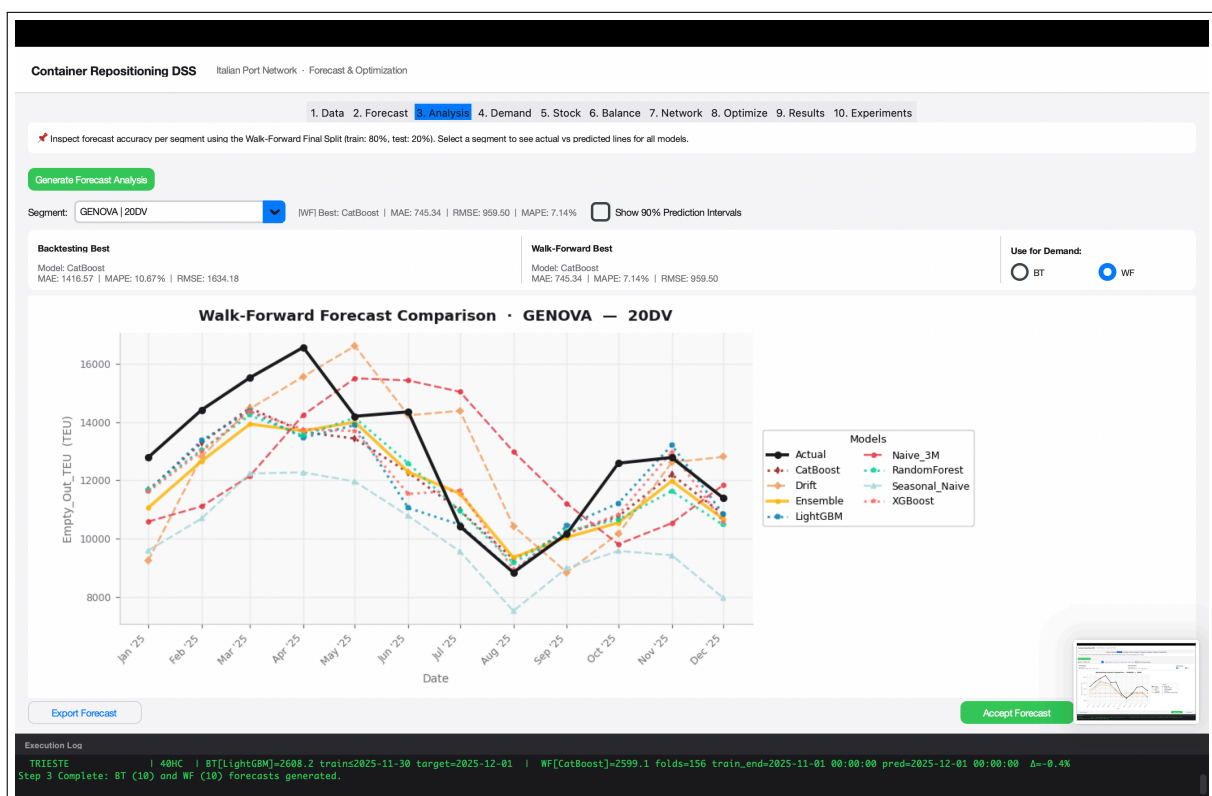


Figure 3.8: Tab 3 – *Analysis*, walk-forward view. Analogous trajectories under the expanding-window walk-forward protocol.

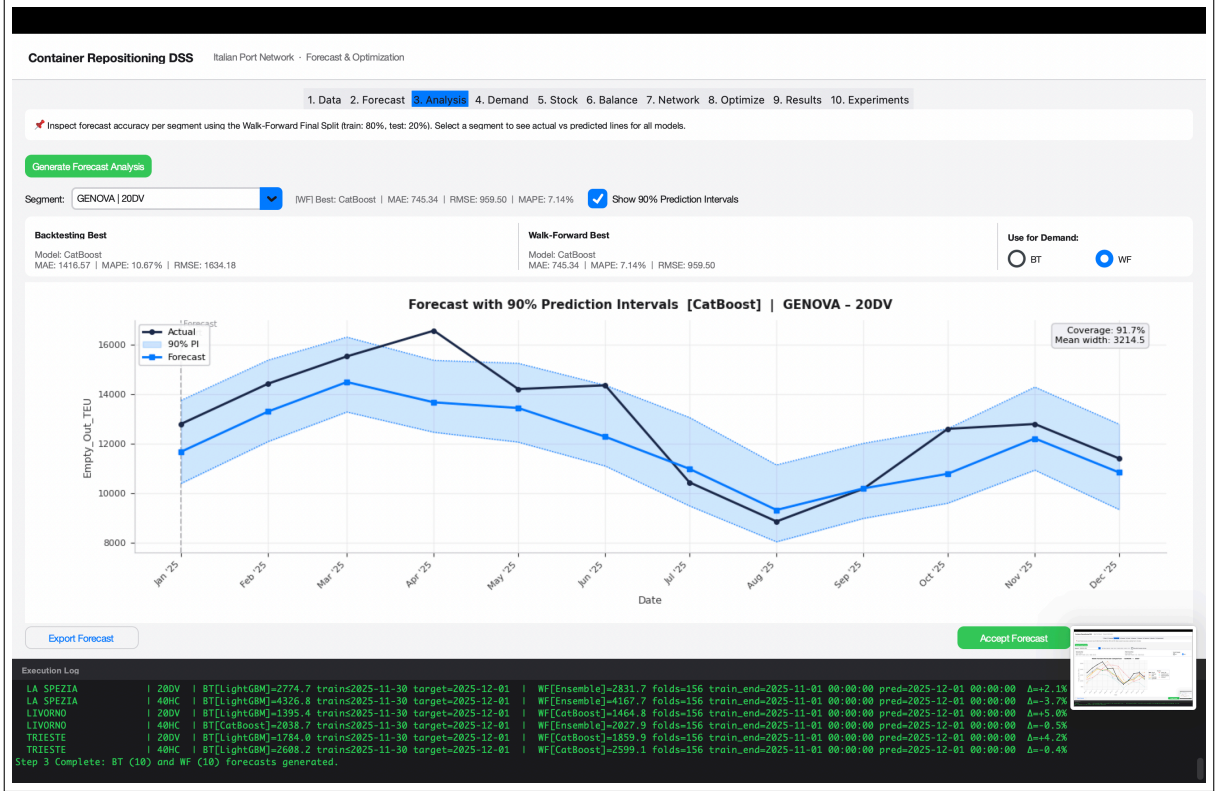


Figure 3.9: Tab 3 – *Analysis*, walk-forward view with 90% prediction intervals enabled. The band is constructed from the 5th and 95th percentiles of the empirical walk-forward residual distribution, and the coverage rate of the realised values inside the band is displayed in the top-right corner as a calibration diagnostic.

3.4 Port-specific demand estimation

The output of the forecasting framework is a single algorithm $m_{j,c}^*$ selected per port–container segment together with its empirical residual distribution. This section translates those segment-level selections into the two quantities that drive the optimisation layer: a point demand forecast at each port (Section 3.4.1) and a set of demand scenarios that capture forecast uncertainty (Section 3.4.2). The two together form the bridge between the predictive and prescriptive layers of the framework.

3.4.1 Point demand forecast

After model selection, the best-in-segment algorithm $m_{j,c}^{*,WF}$ is refitted on the full history through November 2025 and produces a one-step-ahead point forecast $\hat{d}_{j,c}$ for December 2025 demand at port j and container category $c \in \{20DV, 40HC\}$:

$$\hat{d}_{j,c} = \mathbb{E}[y_{j,c,Dec\ 2025} \mid \mathcal{F}_{Nov\ 2025}], \quad (3.13)$$

where $\mathcal{F}_{Nov\ 2025}$ is the information set available up to and including November 2025. The refitting step is essential: although the walk-forward selection of Section 3.3.6 identifies the best algorithm, the pa-

rameters used in that selection are fitted on $\tau_{\min} + 155 = 191$ months of history. Refitting on the full 192 month history adds one observation—the most recent realisation—which matters most under regime shifts when the latest data point carries disproportionate information.

Port-level aggregate demand for December 2025 is the sum of the two container-type forecasts:

$$\hat{D}_j = \hat{d}_{j,20DV} + \hat{d}_{j,40HC}, \quad j \in \mathcal{P}. \quad (3.14)$$

The aggregation is justified because the optimisation layer routes containers as a homogeneous TEU equivalent: the multimodal transport network of Section 3.7 prices each TEU at a mode-specific cost, time, and emission rate, without further disaggregation between the two container categories. The forecasting layer retains the segment-level resolution because demand dynamics differ materially between 20DV and 40HC at the same port; the optimisation layer aggregates because its decisions operate at the TEU level.

The net supply–demand imbalance at each port is then

$$\Delta_j = S_j - \hat{D}_j, \quad j \in \mathcal{P}, \quad (3.15)$$

where S_j is the current stock of empty containers at port j (Section 3.5). Ports with $\Delta_j > 0$ are surplus nodes and act as repositioning origins in the optimisation step; ports with $\Delta_j < 0$ are deficit nodes and act as repositioning destinations. The supply–demand balance $\{\Delta_j\}_{j \in \mathcal{P}}$ together with the network parameters of Section 3.7 determines the feasible region of the optimisation model formulated in Section 3.8.

3.4.2 Empirical residual sampling

Stochastic optimisation requires more than a single point forecast of demand; it requires a set of plausible realisations against which the repositioning plan can be tested. The point forecast $\hat{d}_{j,c}$ produced by the selected algorithm is a best estimate, but the actual realisation will deviate from it, and the size and shape of that deviation is itself information. This section uses that information directly: the historical forecasting errors observed under walk-forward validation tell us, for each segment, how the model has been wrong in the past, and we treat that empirical error distribution as the building block of the demand scenarios.

Residual pool construction

For each port–container segment (j, c) , the walk-forward protocol of Section 3.3.2 yields a pool of $K = 156$ one-step-ahead forecast errors:

$$\varepsilon_{j,c,k} = y_{j,c,t_k+1} - \hat{y}_{j,c,t_k+1}, \quad k = 0, 1, \dots, K-1, \quad (3.16)$$

where the forecasts are produced by the walk-forward best model $m_{j,c}^{*,\text{WF}}$. The pool $\mathcal{R}_{j,c} = \{\varepsilon_{j,c,k}\}_{k=0}^{K-1}$ is the empirical residual distribution of the segment, and the assumptions it encodes are minimal: nothing

is assumed about its parametric form, its symmetry, or its tail behaviour. The framework treats $\mathcal{R}_{j,c}$ as a non-parametric estimator of the forecast-error distribution and uses it directly as the source of scenario randomness.

This is the design choice that separates the present framework from much of the existing stochastic optimisation literature on empty container repositioning. The standard alternative is to fit a parametric distribution—typically a normal—to either the raw demand series or to the residuals, and to draw scenarios from that fitted distribution (Song et al., 2022; Vassos et al., 2025). The standard alternative imposes structure that the empirical residual distribution may not satisfy. Forecast errors in maritime container demand are routinely asymmetric, heavy-tailed, and heteroskedastic, especially over a window that spans the COVID-19 disruption; fitting a Gaussian to such errors smooths away precisely the features that the stochastic optimisation layer is supposed to hedge against. Sampling directly from $\mathcal{R}_{j,c}$ preserves those features.

Scenario construction

A demand scenario is constructed by drawing one error from the residual pool at random (with replacement) and adding it to the point forecast:

$$\tilde{d}_{j,c}^{(s)} = \max(0, \hat{d}_{j,c} + \varepsilon_{j,c}^{(s)}), \quad \varepsilon_{j,c}^{(s)} \sim \text{Unif}(\mathcal{R}_{j,c}), \quad s = 1, \dots, S, \quad (3.17)$$

where $\text{Unif}(\mathcal{R}_{j,c})$ denotes uniform sampling with replacement from the residual pool of segment (j, c) . The procedure generates $S = 30$ scenarios per segment, and each scenario carries equal probability $p_s = 1/S$ in the optimisation model of Section 3.8.

The choice of $S = 30$ is a deliberate compromise between scenario fidelity and solver tractability. Too few scenarios under-represent the residual distribution and produce noisy expected-cost estimates; too many scenarios inflate the size of the two-stage stochastic programme without delivering a proportional improvement in calibration. The sensitivity of the empirical results to this choice is reported in Chapter 5 through a sensitivity study that re-solves the model under $S \in \{10, 20, 30, 50, 100\}$.

Two properties of Eq. (3.17) are worth flagging. First, the $\max(\cdot, 0)$ floor blocks any negative demand draws: a sufficiently negative residual added to a small point forecast can numerically produce a negative demand, which has no physical interpretation. The floor truncates such draws at zero rather than discarding them, which preserves the empirical probability mass at the lower boundary. Second, sampling with replacement preserves the full asymmetry and heavy tails of the realised forecast errors. A bootstrap-style sample without replacement would shrink the effective scenario set as the pool is exhausted; sampling with replacement guarantees that all S draws come from the same distribution and that rare-but-realised residuals can appear more than once in the scenario set.

Limits of the construction

Two practical limits of this construction are worth naming explicitly, because they constrain the operational interpretation of the scenarios and motivate the extensions outlined in Chapter 7.

Independence across time. Residuals are drawn independently across time, so any temporal autocorrelation between consecutive forecast errors is not preserved in the scenarios. For a one-step optimisation horizon this is a mild restriction: each scenario describes a single month, and month-to-month dependence is irrelevant to the first-stage decision. The restriction would become binding under a multi-period rolling-horizon extension, in which residual autocorrelation would propagate through successive planning months. A block-bootstrap variant that resamples contiguous residual blocks is one natural extension and is left for future work.

Independence across ports. Residuals are also sampled independently across ports, so any cross-port co-movement of forecast errors is not preserved either. This is a more substantive restriction. During a system-wide shock such as the COVID-19 disruption, forecast errors tend to be positively correlated across the network: when demand overshoots at Genova, it is likely to overshoot at La Spezia as well. Independent draws by port under-represent this co-movement and therefore under-estimate the tail-risk that the CVaR constraint of Section 3.8 is designed to bound. A multivariate residual model that preserves the empirical cross-port covariance would tighten the tail-risk estimates, and is discussed as a natural extension in Chapter 7.

Both limits are deliberate simplifications that match the single-period scope of the framework as currently formulated. They do not undermine the methodological contribution of grounding scenario construction in empirical residuals rather than in parametric assumptions; they qualify the scope within which that contribution holds.

3.4.3 Demand review in the decision-support platform

The demand step is operationalised in the fourth tab of the decision-support platform (Figure 3.10). The platform pre-fills the table with the segment-level forecasts produced by the analysis step (Section 3.3.7) and exposes seven columns per row: the port, the container type, the back-testing best model and its forecast, the walk-forward best model and its forecast, the realised demand from the calibrated panel (Actual Demand), and the Final Demand Used that the optimiser will consume.

Three aspects of this tab implement the framework’s auditability principle. First, the BT Forecast and WF Forecast columns are displayed side by side rather than the user being asked to pick one in isolation; the dual-protocol selection of Eq. (3.12) therefore remains visible all the way to the optimisation interface. Second, the Actual Demand column—available because the December 2025 realisation is present in the calibrated panel—is shown alongside the two forecasts, so that the user can see the size and direction of the forecast error before committing to a value. Third, the Final Demand Used column is editable through the *Edit Selected Row* panel at the bottom of the tab. The default value mirrors the walk-forward forecast, but the analyst can override it on a per-segment basis when external information not present in the calibrated dataset—a known seasonal disruption, a scheduled vessel call, a contract change—should be reflected in the planning input. The override is logged in the execution log at the bottom of the screen, so that the audit trail records every manual edit alongside the automatic forecasts.

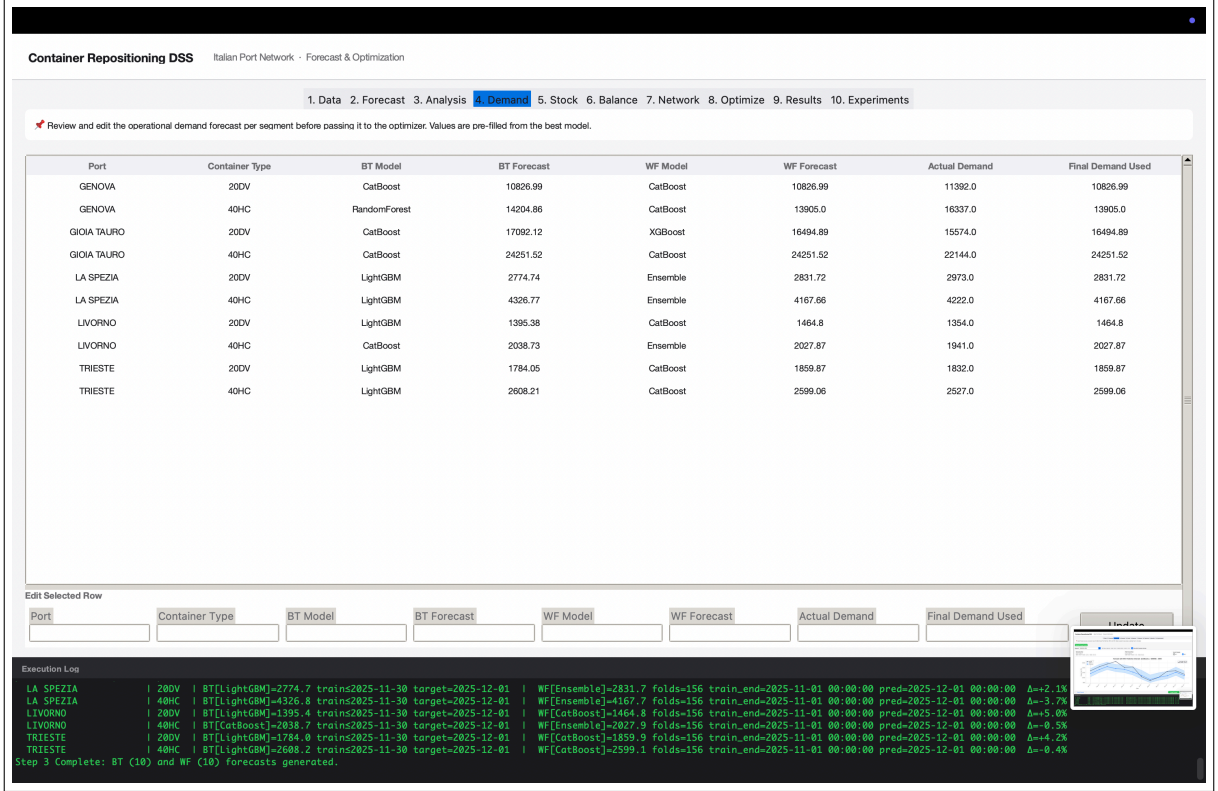


Figure 3.10: Tab 4 – *Demand*. Segment-level demand review table showing the back-testing and walk-forward best models and their forecasts side by side, the realised demand from the calibrated panel, and the Final Demand Used that the optimiser will consume. The bottom panel allows per-row manual override when external information should be reflected in the planning input.

3.5 Stock initialisation

The optimisation layer requires an opening stock S_j of empty containers at each port j at the start of the planning month. This quantity is the supply against which the demand forecast $\hat{D}_j$ of Section 3.4.1 is matched, and it determines whether each port enters the planning step as a surplus or a deficit node. The framework treats the opening stock as a known, non-stochastic input: the single-period formulation does not jointly optimise over stock acquisition, leasing, or inter-period inventory carry-over. The stochastic dimension of the problem is concentrated entirely in the demand realisation, which is the operationally dominant source of uncertainty over a one-month planning horizon.

The platform supports two complementary modes for populating $\{S_j\}_{j \in \mathcal{P}}$. In *automatic* mode, the opening stock is derived from the calibrated panel by accumulating the net empty-flow balance over the trailing months and projecting it forward to the start of the planning window. This auto-extracted vector reflects the structural imbalances documented in the dataset and provides a self-contained default that requires no exogenous information. In *manual* mode, the analyst overrides the auto-extracted values with externally provided stock counts—typically read from terminal yard inventories or carrier equipment-management systems—which overwrite the automatic defaults on a per-port basis. The two modes coexist: the automatic vector is computed unconditionally and remains visible in the Auto Stock

column, while the **Final Stock Used** column records the values that the optimiser will consume. The dual display preserves the audit trail and exposes any operational override alongside the data-driven baseline.

The platform also exposes a small set of preset imbalance regimes through the **Mode** dropdown—*Balanced*, *Moderate Imbalance*, and *Severe Imbalance*—that scale the auto-extracted vector to test the optimiser under different operating conditions. The *Balanced* preset rescales stock proportionally so that the network-wide supply matches the network-wide demand exactly, eliminating the aggregate imbalance and isolating the routing problem. The *Moderate Imbalance* and *Severe Imbalance* presets amplify the empirical imbalance by predefined factors, which is useful for stress-testing the recourse channel and the CVaR constraint formulated in Section 3.8. The default mode used throughout the computational study is *Moderate Imbalance*, which preserves the structural imbalances of the calibrated dataset without amplifying them.

Figure 3.11 shows the stock initialisation tab populated under the *Moderate Imbalance* preset. The five ports are listed in the left column together with their auto-extracted and final stock values; the *Edit Selected Row* panel at the bottom of the tab supports per-port manual override of the final value, with the override logged in the execution log at the bottom of the screen.

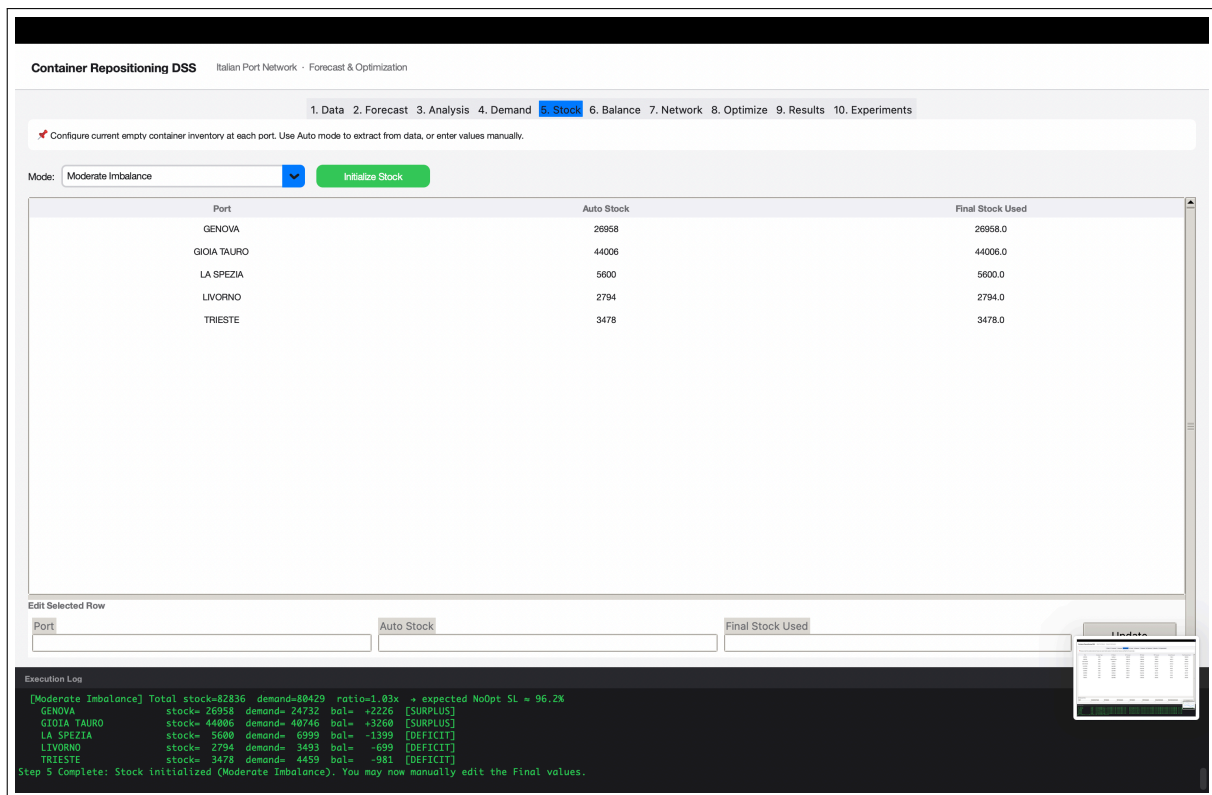


Figure 3.11: Tab 5 – *Stock*. Opening empty-container inventory at each port, populated automatically from the calibrated panel under the *Moderate Imbalance* preset and editable per port through the *Edit Selected Row* panel. The **Auto Stock** column preserves the data-driven baseline while the **Final Stock Used** column records the value that the optimiser will consume.

3.6 Supply-demand balance

With the demand forecast $\hat{D}_j$ from Section 3.4.1 and the opening stock S_j from Section 3.5 in hand, the supply–demand balance step computes the net imbalance Δ_j defined in Eq. (3.15) for each port and classifies the port as surplus ($\Delta_j > 0$), deficit ($\Delta_j < 0$), or balanced ($\Delta_j \approx 0$). This classification is the direct input to the optimisation step: surplus ports act as origins in the repositioning flow, deficit ports act as destinations, and balanced ports neither send nor receive in the first stage of the optimisation.

Two quantities derived from this classification matter for the downstream planning problem. The first is the *aggregate surplus* $\sum_{j:\Delta_j>0} \Delta_j$, which is the total volume of empty containers available for outbound repositioning across the network. The second is the *aggregate deficit* $\sum_{j:\Delta_j<0} |\Delta_j|$, which is the total volume of empty containers required at the destination ports. When the aggregate surplus strictly exceeds the aggregate deficit, the network is *physically feasible*: the optimiser is in principle able to satisfy every destination demand from the available origin supply. When the aggregate deficit exceeds the aggregate surplus, the network is in *structural shortage*: even an optimal repositioning plan must leave some demand unmet, and the optimiser’s role narrows to choosing which shortages to accept and which to cover through emergency recourse. The empirical balance for the December 2025 planning month is reported in Chapter 5, where the network is found to be physically feasible with a small system-wide surplus.

The balance step is operationalised in the sixth tab of the decision-support platform (Figure 3.12). The upper panel of the tab lists the five ports together with their supply, demand, and net balance, colour-coded green for surplus and red for deficit so that the spatial pattern of the imbalance is visible at a glance. The lower panel visualises the same information as a side-by-side bar chart that contrasts current stock against forecast demand at each port; the visual contrast exposes the relative magnitude of each port’s imbalance and identifies the dominant origin–destination pairs that the optimiser will have to address. The tab is read-only: it produces no decision variables of its own, but it surfaces the inputs that the optimisation step will consume in a format the planner can audit before committing to the optimisation run.

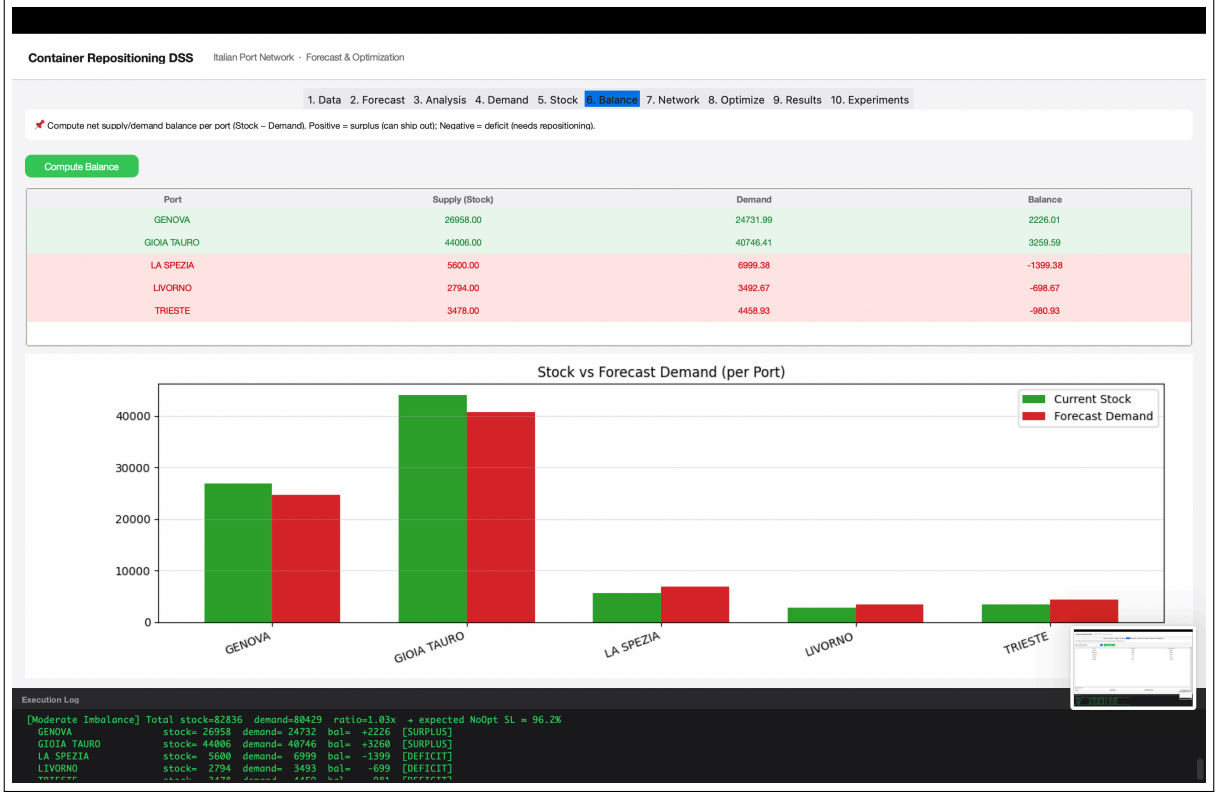


Figure 3.12: Tab 6 – *Balance*. Port-level supply–demand balance for the December 2025 planning month, with surplus ports (Genova, Gioia Tauro) colour-coded green and deficit ports (La Spezia, Livorno, Trieste) colour-coded red in the upper table. The lower panel visualises current stock against forecast demand as a paired bar chart, exposing the spatial pattern of the network imbalance that the optimiser will address.

3.7 Multimodal transport network

The repositioning network connects the five ports of Section 3.2.1 through three transport modes: road haulage, rail haulage, and short-sea shipping. The three modes coexist on every origin–destination corridor in the case-study network, so the optimisation layer of Section 3.8 must trade them against each other on cost, transit time, and CO₂ emissions under capacity constraints that vary by corridor and mode. This section parameterises the network: Section 3.7.1 fixes the topology, Sections 3.7.2 and 3.7.3 specify the per-mode unit parameters and arc capacities, and Section 3.7.5 documents the platform tab that exposes the network to the analyst.

3.7.1 Network topology

Let $\mathcal{P} = \{\text{Genova, Gioia Tauro, La Spezia, Livorno, Trieste}\}$ denote the set of ports. The network includes every ordered origin–destination pair (i, j) with $i, j \in \mathcal{P}$ and $i \neq j$, giving $5 \times 4 = 20$ ordered pairs. Each pair is served by all three transport modes, so the arc set is

$$\mathcal{A} = \{(i, j, m) : i, j \in \mathcal{P}, i \neq j, m \in \{\text{Road, Rail, Sea}\}\}, \quad |\mathcal{A}| = 60. \quad (3.18)$$

The network is therefore directed and complete on the five-port set, with three parallel arcs per ordered pair. The framework does not assume symmetric parameters across reverse arcs: (i, j, m) and (j, i, m) carry separately calibrated cost, time, and emission values, although in practice the two directions tend to share the same per-kilometre rates and differ only on through-traffic effects that the capacity parameter captures.

3.7.2 Per-mode unit parameters

Each arc $(i, j, m) \in \mathcal{A}$ is characterised by five operational parameters: distance d_{ijm} in kilometres, transport cost c_{ijm} in EUR per TEU, transit time τ_{ijm} in hours, CO₂ emissions e_{ijm} in kilograms of CO₂ per TEU, and arc capacity u_{ijm} in TEU per month. The first four are derived from per-kilometre base rates calibrated to established European references; the fifth is calibrated to documented monthly throughput levels on each Italian logistics corridor.

Road haulage. Road haulage is the most flexible mode and is available on every corridor in the network, but its unit cost and emission intensity are the highest of the three modes. The per-kilometre rates are

$$\bar{c}_{\text{Road}} = 1.20 / \text{TEU-km}, \quad e_{\text{Road}} = 0.085 \text{ kg CO}_2 / \text{TEU-km}, \quad (3.19)$$

calibrated from the Italian Ministry of Transport freight cost benchmarks (Ministero delle Infrastrutture e dei Trasporti, 2024) and the EcoTransIT World emission factors (IFEU and INFRAS and IVE, 2024). Road capacity is constrained on long hauls to reflect realistic operational ceilings: a road corridor longer than approximately 200 km carries a markedly smaller monthly throughput than its short-haul equivalent, in line with European hinterland transport practice.

Rail haulage. Rail haulage exploits the electrified Italian rail network and is the lowest-emission option on medium and long corridors. The per-kilometre rates are

$$\bar{c}_{\text{Rail}} = 0.45 / \text{TEU-km}, \quad e_{\text{Rail}} = 0.018 \text{ kg CO}_2 / \text{TEU-km}, \quad (3.20)$$

calibrated from Janić (2007) and the CE Delft European freight emission database (CE Delft, 2020). Rail offers a competitive cost–emission position on medium-distance corridors but is capacity-constrained by the available train slots on each corridor; the capacity parameter u_{ijm} encodes this constraint directly and is the operationally binding limit on rail use throughout the optimisation experiments of Chapter 5.

Short-sea shipping. Short-sea shipping offers the lowest unit cost and the highest capacity on long corridors such as Genova–Gioia Tauro. The per-kilometre rates are

$$\bar{c}_{\text{Sea}} = 0.18 / \text{TEU-km}, \quad e_{\text{Sea}} = 0.025 \text{ kg CO}_2 / \text{TEU-km}, \quad (3.21)$$

calibrated from the Global Logistics Emissions Council framework (Smart Freight Centre, 2023) and the CE Delft database (CE Delft, 2020). The cost advantage of short-sea shipping is most pronounced

on the long southern leg to Gioia Tauro, where its per-TEU cost falls well below the rail equivalent. The emission intensity, by contrast, is slightly above rail on a per-kilometre basis but well below road; under a green-focused objective the optimiser tends to prefer rail on medium corridors and short-sea on long corridors, while road absorbs only the residual flow that cannot be accommodated on the other two modes.

3.7.3 Arc capacities

Arc capacities u_{ijm} are not derived from the per-kilometre rates but are calibrated separately to documented monthly throughput levels on each Italian corridor. The calibration uses three considerations: the documented modal split on each corridor in Italian freight statistics, the available rail train slots and short-sea vessel rotations on the relevant routes, and the operational maximum throughput observed for road on hinterland connections of comparable length. The resulting capacity vector has three properties that matter for the optimisation results.

First, rail capacity is the most binding constraint in the network. Even on corridors where rail is the cheapest and cleanest option—most of the medium-distance pairs—the available train slots exhaust before the demand is fully covered, forcing the optimiser to overflow onto the other two modes. This is the operational reality that the per-kilometre rates alone would mask, and it is the dominant driver of the modal-mix patterns reported in Chapter 5.

Second, short-sea capacity is competitive on long corridors but limited on short ones. The corridor between Genova and La Spezia, for instance, carries a short-sea arc with modest monthly capacity, because the rotation economics of a short coastal call are unfavourable; the same mode carries far higher capacity on the Genova–Gioia Tauro leg, where the longer rotation aligns with an established mainline short-sea service.

Third, road capacity is uniformly high but declines with corridor length: a 110 km road arc (Genova–La Spezia) carries roughly 720 TEU per month, while a 550 km road arc (Genova–Trieste) carries only about 160 TEU per month. This length-dependent cap reflects fleet rotation constraints and is one of the levers through which the network deters the optimiser from defaulting to road on long legs.

3.7.4 Representative corridor: Genova–La Spezia

The combined effect of the per-mode rates and the corridor-specific capacities is illustrated by Table 3.1, which reports the three arc rows for the Genova–La Spezia corridor as a representative port pair. Road covers the shortest distance (110 km) at the highest per-TEU cost (132) and emission load (9.35 kg) but with the largest capacity (720 TEU); rail carries the longest distance of the three modes (105 km, comparable to road) at a much lower cost (47) and emission (1.89 kg) but with a far smaller monthly capacity (291 TEU); short-sea covers 150 km—a coastal detour—at the lowest cost (27) and modest emission (3.75 kg) with intermediate capacity (647 TEU). The trade-off pattern that Table 3.1 exposes—cheaper modes are typically the cleanest but also the most capacity-constrained—generalises across the network and is what makes the optimisation problem non-trivial.

Table 3.1: Multimodal arc parameters for the Genova–La Spezia corridor.

From	To	Mode	Distance	Cost	Time	CO ₂	Capacity
			(km)	(/TEU)	(hr)	(kg/TEU)	(TEU/month)
Genova	La Spezia	Road	110	132	2.5	9.35	720
Genova	La Spezia	Rail	105	47	4.0	1.89	291
Genova	La Spezia	Sea	150	27	8.0	3.75	647

3.7.5 Network configuration in the decision-support platform

The multimodal network is operationalised in the seventh tab of the decision-support platform (Figure 3.13). The tab displays the full $|\mathcal{A}| = 60$ arc rows in a scrollable table, with eight columns per row: From, To, Mode, Distance_km, Cost_per_TEU, Time_hr_per_TEU, CO2_kg_per_TEU, and Cap_TEU. The default network is populated through the *Load Default Routes* button using the calibrated values of Sections 3.7.2 and 3.7.3; the *Edit Selected Row* panel at the bottom of the tab supports per-arc manual override of any of the eight parameters, which is the mechanism through which the analyst can adapt the network to a different cost or capacity scenario without re-deriving the calibration from scratch.

Two design choices in this tab reflect the framework’s reproducibility principle. First, the default network loaded by the button is fixed across all experiments reported in this thesis; the *Load Default Routes* action does not draw from a stochastic process or a user-specific configuration, so two analysts who load the defaults independently see exactly the same arc parameters. Second, every per-arc override is logged in the execution log at the bottom of the screen, so that the audit trail records every deviation from the calibrated baseline alongside the optimisation results that the deviation produces. The combination of fixed defaults and a transparent override mechanism preserves the reproducibility of the baseline experiments while leaving the network configurable for sensitivity studies.

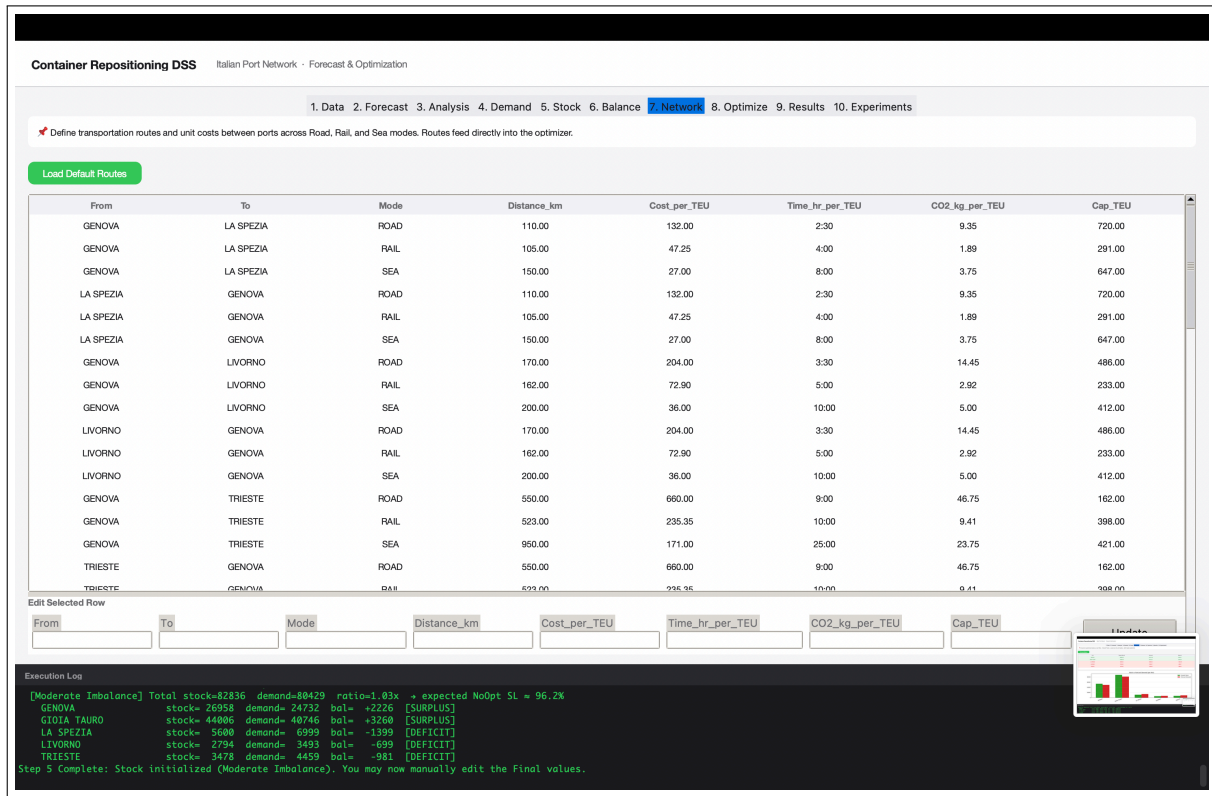


Figure 3.13: Tab 7 – *Network*. Full table of the $|\mathcal{A}| = 60$ multimodal arcs in the case-study network. Each arc row carries distance, per-TEU cost, transit time, CO₂ emission factor, and monthly capacity. The default network is loaded through the *Load Default Routes* button using the calibrated values of Sections 3.7.2 and 3.7.3; per-arc manual override is supported through the *Edit Selected Row* panel at the bottom of the tab.

3.8 Two-stage stochastic optimisation model

The optimisation step takes the demand forecasts and scenarios produced in Section 3.4, the opening stocks of Section 3.5, and the multimodal network of Section 3.7, and produces a repositioning plan that specifies how many empty containers to ship along each arc and which mode to use. The plan is committed before demand uncertainty is resolved, and any residual imbalance that emerges once demand is realised is absorbed through a recourse channel priced at a route-aware shortage penalty. The model is formulated as a two-stage stochastic linear programme with a Conditional Value-at-Risk (CVaR) constraint in the objective, which jointly captures the operationally distinct effects of average cost control and tail-risk control under demand uncertainty.

This section develops the model in five subsections: Section 3.8.1 fixes the sets, indices, and parameters; Section 3.8.2 derives the route-aware recourse penalty that is one of the methodological contributions of the thesis; Section 3.8.4 introduces the decision variables; Section 3.8.5 formulates the objective function together with the multi-objective arc-score normalisation; and Section 3.8.6 lists the constraints, including the linearisation that embeds CVaR within a linear programme.

3.8.1 Sets, indices, and parameters

Let $\mathcal{P}$ denote the set of ports introduced in Section 3.2.1 and $\mathcal{A}$ the set of $|\mathcal{A}| = 60$ directed arcs defined in Eq. (3.18). Demand uncertainty is represented by a finite scenario set $\mathcal{S} = \{1, 2, \dots, S\}$ of cardinality $S = 30$, with scenarios sampled from the empirical walk-forward residual pool of Section 3.4.2. All scenarios carry equal probability $p_s = 1/S$, which preserves the empirical nature of the construction: no parametric weighting is imposed on top of the non-parametric residual distribution.

Each arc $(i, j, m) \in \mathcal{A}$ inherits the five operational parameters fixed in Section 3.7.2: distance d_{ijm} (km), transport cost c_{ijm} (/TEU), transit time τ_{ijm} (hours), CO₂ emissions e_{ijm} (kg/TEU), and arc capacity u_{ijm} (TEU per month). At each port $j \in \mathcal{P}$, the symbol S_j denotes the opening stock of empty containers (Section 3.5) and $\tilde{d}_j^{(s)}$ denotes the demand realisation under scenario s obtained by aggregating the segment-level scenarios of Eq. (3.17):

$$\tilde{d}_j^{(s)} = \tilde{d}_{j,20DV}^{(s)} + \tilde{d}_{j,40HC}^{(s)}, \quad j \in \mathcal{P}, \quad s \in \mathcal{S}. \quad (3.22)$$

The CVaR mechanism is governed by two parameters: a confidence level $\alpha = 0.95$ and an objective weight $\lambda \in [0, 1]$ that controls how strongly the tail-cost term enters the criterion. The confidence level $\alpha = 0.95$ is the conventional choice in risk-aware logistics optimisation: it focuses the CVaR term on the worst 5% of scenarios while leaving the expected-cost term to manage the remaining 95%. The weight λ is set to 0.5 in the baseline configuration, which gives equal nominal weight to the expected-cost and CVaR components; the sensitivity of the results to both α and λ is reported in Chapter 5.

Two cost parameters complete the parameterisation. The *over-shipment* cost $c^- = 50$ /TEU is charged on excess inventory held at a destination port when the realised demand falls below the planned inbound flow; the value represents the storage and demurrage cost of an unwanted container occupying yard space for a planning period. The *under-shipment* cost c_j^+ is charged on unmet demand at destination port j and is constructed in the next subsection.

3.8.2 Route-aware recourse penalty

The treatment of unmet demand in the present framework departs from the standard fixed-penalty convention used in most stochastic empty-container repositioning models. The standard convention is to apply a single penalty coefficient—typically denoted c^+ and set exogenously—uniformly across all destination ports, irrespective of where the shortage occurs in the network. This convention has two well-known disadvantages. First, it is unfaithful to the operational economics of recourse: the actual cost of covering a shortage at one port differs from the cost at another, because the cheapest reachable emergency shipment uses different modes and corridors at each destination. Second, when the uniform penalty is set too low, the optimiser may treat unmet demand as cheaper than routing into a destination port, producing plans that under-supply destinations whose served cost strictly exceeds the shortage penalty. This is not merely a calibration issue but a structural inconsistency that a uniform constant cannot guarantee against in a heterogeneous multi-port network.

The framework addresses this inconsistency by constructing a port-specific, network-consistent re-

course penalty. For each destination $j \in \mathcal{P}$, the under-shipment cost is defined as

$$c_j^+ = \min_{(i,j,m) \in \mathcal{A}} c_{ijm} + \pi, \quad \pi = 30/\text{TEU}, \quad (3.23)$$

where the minimum is taken over all incoming arcs to port j and π is a service-delay surcharge that captures the operational cost accumulating while an emergency shipment is arranged.

Three properties of Eq. (3.23) make the construction operationally faithful. First, the minimum operator $\min_{(i,j,m) \in \mathcal{A}}$ locates the *cheapest reachable* emergency repositioning into port j . The optimiser cannot plead a lower shortage cost than the cost of covering the shortage through the network itself; the recourse channel is therefore at least as expensive as the cheapest planned alternative, which removes the pathological case where unmet demand looks artificially attractive in the objective function. Second, the surcharge π captures the service-delay cost associated with emergency procurement: an emergency shipment is by definition arranged outside the ordinary planning cycle, and the surcharge prices the operational disruption that this imposes on the destination terminal. The value $\pi = 30/\text{TEU}$ is calibrated to the order of magnitude of documented emergency repositioning premiums in Mediterranean container logistics and represents the marginal cost above the cheapest reachable route. Third, c_j^+ is *port-specific*: a deep deficit port served only by long-haul corridors carries a higher recourse penalty than a port served by short-haul rail, reflecting the genuine difference in emergency-cover economics between the two ports.

The route-aware recourse penalty is one of the methodological contributions of this thesis (Section 1.6). It changes the behaviour of the optimiser in two non-trivial ways. First, the geographic distribution of unmet demand across the network reflects the relative cost of covering shortages at each destination, rather than being driven by a single uniform constant that has no geographic content. Second, the comparison between deterministic and stochastic plans becomes methodologically clean: the deterministic plan is charged at the same c_j^+ schedule as the stochastic plan, so any cost advantage of the stochastic plan reflects genuine value of the two-stage formulation rather than an accidental artefact of how shortages were priced.

3.8.3 Multi-objective configuration of arc costs

The routing component of the objective function (Section 3.8.5) is driven by a *normalised arc score* that places the three operational criteria—cost, transit time, and CO₂ emissions—on a common $[0, 1]$ scale. For each arc $(i, j, m) \in \mathcal{A}$, the normalised arc score is

$$\bar{c}_{ijm} = \frac{w_C c_{ijm}}{\max_{(i,j,m) \in \mathcal{A}} c_{ijm}} + \frac{w_T \tau_{ijm}}{\max_{(i,j,m) \in \mathcal{A}} \tau_{ijm}} + \frac{w_E e_{ijm}}{\max_{(i,j,m) \in \mathcal{A}} e_{ijm}}, \quad (3.24)$$

where $w_C, w_T, w_E \geq 0$ are user-specified weights and the three denominators are the maximum values of cost, time, and emission observed across the arc set. Normalisation by the maximum is essential: the three criteria have incommensurable units (/TEU, hours, kg/TEU) and cover different numerical ranges, so a naive weighted sum would let the criterion with the largest absolute scale dominate the score regardless of the analyst's intent. After normalisation, the weights (w_C, w_T, w_E) reflect planning priorities directly and are interpretable as relative importances rather than as physical unit conversions.

The framework studies three weight configurations that correspond to operationally distinct planning postures.

Cost-focus. Under the cost-focus configuration, the weights are $(w_C, w_T, w_E) = (0.75, 0.10, 0.15)$, which assigns dominant weight to cost and treats time and emissions as secondary constraints. The optimiser under this configuration routes traffic to the cheapest feasible mode at each origin–destination pair, which is short-sea on long corridors and rail on medium corridors. The configuration corresponds to a status-quo carrier perspective that prices decarbonisation only weakly.

Balanced. Under the balanced configuration, the weights are $(w_C, w_T, w_E) = (1/3, 1/3, 1/3)$, which treats the three criteria symmetrically. The optimiser under this configuration is the natural neutral reference for the Pareto analysis of Chapter 5: it places no operating point on the cost-optimal or the green-optimal corner of the frontier, and it is therefore the benchmark against which the cost premium of greener routing is measured.

Green-focus. Under the green-focus configuration, the weights are $(w_C, w_T, w_E) = (0.10, 0.10, 0.80)$, which assigns dominant weight to emissions and treats cost and time as secondary constraints. The optimiser under this configuration penalises emission-intensive routing and favours rail over sea on most corridors and either rail or sea over road wherever capacity permits. The configuration corresponds to a decarbonisation-aware planning posture that internalises the emission externality at a high implicit shadow price.

The three weight configurations are not the only ones the framework supports—any non-negative triple on the unit simplex is admissible—but they span the operationally relevant range and are used as the three end-to-end variants in the experimental design of Chapter 4.

A final clarification is operationally important. The normalised arc score $\bar{c}_{ijm}$ governs the *routing decision* alone, not the monetary accounting of the plan. The optimiser uses $\bar{c}_{ijm}$ to choose which arcs to load and in what proportion; the monetary key performance indicators reported in Chapter 5 are recovered *ex post* from the raw arc costs c_{ijm} and the recourse rates c_j^+ and c^- , on a consistent EUR scale. The separation between the routing criterion (normalised, multi-objective) and the reporting metric (raw, monetary) keeps the trade-offs explicit at the planning interface while preserving a single accounting standard at the results interface.

3.8.4 Decision variables

The two-stage stochastic programme defines two classes of decision variables that differ in the information set under which they are committed. The first-stage variable represents the routing commitment that must be fixed before demand uncertainty is resolved. The second-stage variables represent the recourse actions that absorb the residual imbalance once each scenario is realised. A third group of auxiliary variables implements the CVaR linearisation that embeds the tail-risk constraint within the linear programme.

First-stage routing variable. For each arc $(i, j, m) \in \mathcal{A}$, the first-stage variable

$$x_{ijm} \geq 0 \quad (3.25)$$

denotes the number of empty containers committed to arc (i, j, m) before demand is observed. The collection $\{x_{ijm}\}_{(i,j,m) \in \mathcal{A}}$ is the *repositioning plan*—the operational output of the optimisation step that a port planner would implement at the start of the planning month. Because the variable is committed before any scenario is realised, it does not carry a scenario index: the plan is robust by construction across the full scenario set $\mathcal{S}$, with all scenario-dependent adjustments delegated to the second-stage variables.

Second-stage recourse variables. For each destination port $j \in \mathcal{P}$ and each scenario $s \in \mathcal{S}$, two non-negative recourse variables react to the realised demand:

$$y_j^{+, (s)} \geq 0, \quad (\text{under-shipment at port } j \text{ under scenario } s) \quad (3.26)$$

$$y_j^{-, (s)} \geq 0. \quad (\text{over-shipment at port } j \text{ under scenario } s) \quad (3.27)$$

The variable $y_j^{+, (s)}$ is the quantity of unmet demand at port j under scenario s , which has to be covered through emergency recourse at the route-aware penalty c_j^+ defined in Eq. (3.23). The variable $y_j^{-, (s)}$ is the quantity of leftover stock held at port j under scenario s as excess inventory, incurring the storage cost c^- defined in Section 3.8.1. The two variables enforce the scenario-wise demand balance through Constraint (3.32) below: under each scenario, the planned inbound flow plus the under-shipment minus the over-shipment must equal the realised demand. Only one of the two variables can take a strictly positive value at the optimum: a port that is under-supplied has $y_j^{+, (s)} > 0$ and $y_j^{-, (s)} = 0$, while a port that is over-supplied has the opposite. Both being positive would be sub-optimal because the over-shipment penalty c^- and the under-shipment penalty c_j^+ are both strictly positive and the optimiser would reduce both by cancelling them against each other.

CVaR auxiliary variables. Two auxiliary variables implement the Rockafellar–Uryasev linearisation of the CVaR term in the objective (Rockafellar and Uryasev, 2000). The first is a scalar Value-at-Risk threshold,

$$\eta \in \mathbb{R}, \quad (3.28)$$

which is unrestricted in sign. At the optimum of the linearised problem, η takes the value of the Value-at-Risk at confidence level α : that is, the smallest cost threshold such that the probability of total cost exceeding η is at most $1 - \alpha$. The second is a scenario-indexed slack,

$$z_s \geq 0, \quad s \in \mathcal{S}, \quad (3.29)$$

which captures the amount by which the total monetary cost under scenario s exceeds the threshold η . The linearisation that ties these auxiliaries to the total-cost distribution is given in Constraint (3.35) below.

The two CVaR auxiliaries are mathematical constructs, not operational decisions: η is a threshold that the optimiser chooses to balance the expected-cost term against the tail-cost term, and z_s is a slack that activates only on the scenarios in the right tail of the cost distribution. They do not appear in the repositioning plan that the analyst implements, but they are essential to making the CVaR constraint solvable as a linear programme rather than as a non-convex problem in the original risk-measure form.

3.8.5 Objective function

The optimisation model minimises a weighted sum of three terms: the first-stage routing cost, the expected second-stage recourse cost, and the CVaR-based tail-risk term. The full objective is

$$\begin{aligned} \min_{x, y^+, y^-, \eta, z} \quad & \underbrace{\sum_{(i,j,m) \in \mathcal{A}} \bar{c}_{ijm} x_{ijm}}_{\text{first-stage routing cost}} + \underbrace{\sum_{s \in \mathcal{S}} p_s \sum_{j \in \mathcal{P}} \left(c_j^+ y_j^{+, (s)} + c_j^- y_j^{-, (s)} \right)}_{\text{expected recourse cost}} \\ & + \underbrace{\lambda \left(\eta + \frac{1}{1 - \alpha} \sum_{s \in \mathcal{S}} p_s z_s \right)}_{\text{CVaR}_\alpha \text{ tail-risk term}}. \end{aligned} \quad (3.30)$$

Each term in Eq. (3.30) represents an operationally distinct cost component. The *first-stage routing cost* is the normalised score of every loaded arc, summed across the network. It is driven by the multi-objective arc score $\bar{c}_{ijm}$ of Eq. (3.24), which combines cost, time, and emissions on a common $[0, 1]$ scale under the analyst-specified weight configuration. The monetary equivalent of this term, together with the time and emission accountings, is recovered *ex post* from the raw arc parameters of Section 3.7.2 and reported as the corresponding key performance indicators in Chapter 5.

The *expected recourse cost* is the probability-weighted sum of the second-stage costs across the scenario set. For each scenario s , the inner sum aggregates the under-shipment and over-shipment costs across all destination ports, and the outer sum weights each scenario by its probability $p_s = 1/S$. Because the under-shipment cost c_j^+ is set port by port through Eq. (3.23), the recourse term charges shortages at the cost of covering them through the cheapest reachable network arc plus the service-delay surcharge. This is the operational channel through which the recourse layer absorbs the forecast-error volume that the first-stage plan cannot anticipate.

The *CVaR tail-risk term* is the linearised Conditional Value-at-Risk of the total monetary cost across the scenario set, weighted by λ . The formulation follows Rockafellar and Uryasev (2000): CVaR_α is expressed as the minimum over η of $\eta + (1 - \alpha)^{-1} \mathbb{E}[\max(0, \text{Cost} - \eta)]$, and the inner maximum is linearised by introducing the non-negative slack variables z_s defined in Eq. (3.29). At the optimum, η takes the value of the α -Value-at-Risk and the slack z_s takes the positive part of the excess cost in scenario s , so the term $\eta + (1 - \alpha)^{-1} \sum_s p_s z_s$ equals CVaR_α of the cost distribution. This construction admits a linear-programming formulation without resorting to chance constraints or mixed-integer reformulations (Rockafellar and Uryasev, 2000).

Three properties of this objective deserve emphasis. First, the three terms are operationally non-substitutable. The first-stage cost prices routing choices, the expected recourse cost prices the average

miss, and the CVaR term prices the right tail of the cost distribution; reducing one does not directly substitute for reducing another. A first-stage plan that fully eliminates routing cost (by sending nothing) inflates the recourse cost and the CVaR; a plan that eliminates the expected miss (by over-supplying every destination) inflates the routing cost; and a plan that bounds the worst-case tail tightly (by paying a large insurance premium) inflates the expected cost. The optimiser balances all three under the weight λ that the analyst specifies.

Second, the CVaR term enters the objective with a *linear* weight λ rather than as a threshold constraint. This is a design choice. The alternative formulation imposes $\text{CVaR}_\alpha \leq \bar{c}$ for some predetermined tail-cost ceiling $\bar{c}$, which is operationally natural when the planner has a contractual or regulatory cap on tail exposure. The weight-based formulation chosen here is more flexible: it lets the analyst vary λ from 0 (risk-neutral expected-cost minimisation) to large values (strongly risk-averse) and trace the resulting operating points along the cost–CVaR frontier, which is reported in the sensitivity study of Chapter 5.

Third, the routing weights (w_C, w_T, w_E) and the CVaR weight λ are formally independent controls. The first triple steers the multi-objective trade-off between cost, time, and emissions in the first-stage routing; the scalar λ steers the trade-off between average and tail performance in the second stage. In the experiments of Chapter 4, the two controls are varied independently to disentangle the contribution of green routing from the contribution of tail-risk control: the 9% green-routing premium and the 12% CVaR premium reported in Chapter 5 are estimated through pairwise comparisons that hold one control fixed while moving the other.

3.8.6 Constraints

The two-stage stochastic programme is completed by six groups of constraints. Constraints (3.31)–(3.34) govern the first-stage repositioning plan, Constraint (3.32) couples the first stage to the second-stage recourse variables under each demand scenario, and Constraints (3.35)–(3.36) implement the CVaR linearisation and enforce non-negativity of the recourse and slack variables. The full system reads as follows:

$$\sum_{\substack{j \in \mathcal{P}, m \in \{\text{Road, Rail, Sea}\} \\ (i, j, m) \in \mathcal{A}}} x_{ijm} \leq S_i \quad \forall i \in \mathcal{P}, \quad (3.31)$$

$$\sum_{\substack{i \in \mathcal{P}, m \in \{\text{Road, Rail, Sea}\} \\ (i, j, m) \in \mathcal{A}}} x_{ijm} + y_j^{+, (s)} - y_j^{-, (s)} = \tilde{d}_j^{(s)} \quad \forall j \in \mathcal{P}, s \in \mathcal{S}, \quad (3.32)$$

$$x_{ijm} \leq u_{ijm} \quad \forall (i, j, m) \in \mathcal{A}, \quad (3.33)$$

$$x_{ijm} \geq 0 \quad \forall (i, j, m) \in \mathcal{A}, \quad (3.34)$$

$$\sum_{(i, j, m) \in \mathcal{A}} c_{ijm} x_{ijm} + \sum_{j \in \mathcal{P}} (c_j^+ y_j^{+, (s)} + c^- y_j^{-, (s)}) - \eta - z_s \leq 0 \quad \forall s \in \mathcal{S}, \quad (3.35)$$

$$y_j^{+, (s)}, y_j^{-, (s)}, z_s \geq 0 \quad \forall j \in \mathcal{P}, s \in \mathcal{S}. \quad (3.36)$$

The six groups deserve separate treatment because each encodes a distinct operational requirement.

Supply constraints (3.31). The supply constraint limits the total outbound repositioning from each origin port to its available empty-container stock. The inner sum aggregates the first-stage flow across all destinations and all modes leaving port i , and the right-hand side is the opening stock S_i determined in Section 3.5. The constraint is enforced as an inequality rather than an equality: a port is permitted to retain part of its surplus stock if shipping the full surplus would not be advantageous under the objective function. In practice this slack matters only at balanced or surplus ports near the margin; at the deeper surplus nodes the optimiser typically exhausts the available stock through the cheapest reachable routing.

Demand balance (3.32). The demand-balance constraint is the coupling that links the first and second stages of the programme. For every destination port j and every scenario s , the realised demand $\tilde{d}_j^{(s)}$ must be matched by the sum of the planned inbound flow—the first-stage variable x_{ijm} aggregated over all origins and modes—plus the under-shipment recourse $y_j^{+, (s)}$ minus the over-shipment recourse $y_j^{-, (s)}$. The constraint must hold as an equality, which is what forces the recourse variables to absorb every imbalance scenario by scenario. If the planned inbound flow exceeds the realised demand, the over-shipment recourse takes the excess and charges c^- per TEU; if the planned inbound flow falls short of realised demand, the under-shipment recourse covers the shortage and charges the port-specific recourse penalty c_j^+ derived in Eq. (3.23). The strict equality is essential because the recourse mechanism is what absorbs the realised forecast error—a relaxed equality would let the model implicitly drop demand without paying any penalty.

Capacity constraints (3.33). The capacity constraint bounds the first-stage flow on each arc by its monthly capacity u_{ijm} calibrated in Section 3.7.3. This is an arc-level constraint rather than a corridor-level constraint: each of the three modes serving an origin–destination pair has its own capacity, so the total flow on the corridor can exceed the capacity of any single mode by distributing across modes. The

binding cases reported in Chapter 5 arise predominantly on rail arcs, which is the operational consequence of the calibration outlined in Section 3.7.3: rail offers the best cost–emission trade-off on medium corridors but exhausts its capacity well before the demand is covered, forcing overflow onto road or short-sea arcs.

Non-negativity of routing (3.34). The first-stage routing variable is required to be non-negative. Negative flows have no physical interpretation in the repositioning context: a container cannot be “un-routed” through an arc, so allowing negative x_{ijm} would admit purely mathematical solutions with no operational counterpart.

CVaR linearisation (3.35). The CVaR linearisation constraint implements the Rockafellar–Uryasev formulation (Rockafellar and Uryasev, 2000) that embeds the tail-risk term within the linear programme. For every scenario s , the constraint requires that the total monetary cost of scenario s —the first-stage routing cost charged at the raw arc costs c_{ijm} , plus the scenario-specific recourse cost—does not exceed the sum of the Value-at-Risk threshold η and the non-negative slack z_s . The slack therefore takes the positive part of the excess cost above the threshold, which is exactly what the CVaR term in the objective requires. Three properties of this constraint matter operationally.

First, the constraint uses the *raw* arc cost c_{ijm} , not the normalised arc score $\bar{c}_{ijm}$ of Eq. (3.24). The distinction is deliberate. The routing decision in the first-stage term of the objective is driven by the normalised score, which reflects the analyst’s multi-objective weights; the monetary cost in the CVaR linearisation, however, is the genuine EUR cost of the scenario, which must be measured on the underlying monetary scale for the tail-risk bound to be operationally meaningful. The separation between the normalised routing criterion and the monetary reporting criterion, introduced in Section 3.8.3, is what makes this two-scale formulation coherent.

Second, the constraint is written with a single η for all scenarios rather than scenario-specific thresholds. This is essential to the Rockafellar–Uryasev linearisation: η is the single Value-at-Risk threshold against which all scenarios are measured, and the slack z_s activates only on those scenarios whose monetary cost exceeds η . A scenario whose cost is below η takes $z_s = 0$, contributing nothing to the CVaR term in the objective.

Third, the constraint admits the recourse rates c_j^+ and c^- inside its monetary accounting. The CVaR tail-risk bound therefore covers the cost of recourse alongside the cost of first-stage routing, which is the operationally correct accounting: a plan whose worst-case tail is driven by emergency recourse should be priced at the recourse rate in the tail-risk term, not at the routing rate.

Non-negativity of recourse and slack (3.36). The under-shipment and over-shipment recourse variables, together with the CVaR slack, are required to be non-negative. Each of the three quantities has a clear operational interpretation that does not admit negative values: an under-shipment cannot exceed zero in the negative direction (a port cannot have negative unmet demand), an over-shipment is likewise non-negative (excess inventory is by definition non-negative), and the CVaR slack measures the positive part of the excess cost above the threshold and is non-negative by construction.

The complete model (3.30)–(3.36) is a linear programme in $|\mathcal{A}| + 2 \cdot |\mathcal{P}| \cdot S + S + 1$ continuous variables and $|\mathcal{P}| + |\mathcal{P}| \cdot S + |\mathcal{A}| + S$ inequality and equality constraints, ignoring sign restrictions. For the present case study with $|\mathcal{P}| = 5$, $|\mathcal{A}| = 60$, and $S = 30$, this gives $60 + 300 + 30 + 1 = 391$ continuous variables and $5 + 150 + 60 + 30 = 245$ non-trivial constraints. The instance is well within the tractable size for commodity linear-programming solvers and solves to optimality in a fraction of a second on standard hardware, which is essential to the deployment feasibility of the platform documented in the next section.

3.9 Decision-support platform architecture

The methodological framework formalised in Sections 3.1–3.8.6 is operationalised as a working decision-support platform. The platform is the second deliverable of this thesis: it instantiates the framework as software that an analyst can actually run, audit, and extend, rather than as a paper specification that requires re-implementation from scratch at every deployment site. This section documents the platform architecture, the software stack, and the dual-branch execution model that distinguishes it from a single-pipeline implementation, and concludes with the operational detail of the optimisation tab that ties the foregoing mathematical model to the user interface.

3.9.1 Modular architecture and software stack

The platform is implemented in Python and organised around the principle that each pipeline stage should be independently auditable. The internal organisation mirrors the methodological structure of this chapter: each major methodological component is implemented as a separate module, and the modules communicate through explicit data interfaces—pandas DataFrames for tabular inputs and outputs, plain Python dictionaries for run-level metadata—rather than through hidden side effects on a global state. The result is a platform whose components can be modified, replaced, or extended without rewriting the surrounding pipeline; an analyst who wishes to swap in a new forecasting algorithm, an additional transport mode, or a different risk measure can do so by replacing the corresponding module while leaving the rest of the pipeline untouched.

The software stack is deliberately minimal and built on widely supported scientific Python libraries. The data layer relies on pandas for tabular manipulation and numpy for numerical operations. The forecasting layer uses scikit-learn for the Random Forest implementation and the gradient-boosting libraries xgboost, lightgbm, and catboost for the three boosting variants, with statsmodels providing the Diebold–Mariano test implementation and supporting diagnostic routines. The optimisation layer is built on PuLP, a Python interface to mixed-integer linear-programming solvers, configured to call CBC as the default open-source backend. The visualisation layer uses matplotlib for the static charts embedded in the platform’s tabs and networkx for the repositioning-network visualisation rendered in the results tab. The user interface itself is built on tkinter, the standard Python GUI library, which keeps the platform deployable on commodity hardware without requiring a web server, a database engine, or any external service.

The dependency on open-source components rather than proprietary solvers is a deliberate choice. The framework is intended to be reproducible on any machine that can run a standard Python distribution, which excludes pipelines that depend on commercial solver licences. The trade-off is solver performance: a commercial solver such as Gurobi or CPLEX would solve the linear programme of Section 3.8.6 marginally faster, but the open-source backend is sufficient for the problem size at hand and preserves the platform’s portability.

3.9.2 ten-tab user interface

The platform exposes the methodological pipeline through a ten-tab interface organised into four logical phases. The four phases mirror the analyst’s workflow rather than the internal module structure: they group together the steps that an analyst performs in sequence at the start of each planning cycle and separate them from the steps that the analyst revisits less frequently.

- **Phase 1 – Data ingest** (Tab 1, documented in Section 3.2.6). Loads the calibrated panel, validates the schema, and exposes a preview of the raw flows for analyst audit.
- **Phase 2 – Forecast and evaluation** (Tabs 2–4, documented in Sections 3.3.3, 3.3.7, and 3.4.3). Executes the back-testing and walk-forward evaluation protocols, runs the pairwise Diebold–Mariano tests, surfaces the per-segment forecast comparison with optional prediction intervals, and allows the analyst to commit a final demand forecast with optional manual overrides.
- **Phase 3 – Optimisation set-up** (Tabs 5–7, documented in Sections 3.5, 3.6, and 3.7.5). Initialises the opening stock, computes the supply–demand balance, and parameterises the multimodal network through the 60-arc table.
- **Phase 4 – Optimisation, results, and diagnostics** (Tabs 8–12, documented in the remainder of this chapter). Executes the two-stage stochastic optimisation, visualises the resulting repositioning plan, runs the experimental scenarios and the sensitivity studies, and traces the Pareto frontier between cost and CO₂ emissions.

The ten-tab structure is intentionally linear: the phases are visited in order during a normal planning cycle, with later tabs locked until the prerequisite tabs have been committed. The locking mechanism is soft—an analyst can revisit any tab once it has been unlocked—but the default forward flow prevents common operator errors such as running the optimiser before the demand forecast has been accepted.

3.9.3 Dual-branch BT/WF execution

A distinctive feature of the platform is the *dual-branch* execution model that runs the back-testing (BT) and walk-forward (WF) pipelines in parallel and surfaces both estimates side by side at every step of the analysis. The motivation is the methodological asymmetry between the two protocols identified in Section 3.3.2: back-testing offers a familiar single-window diagnostic that is easy to interpret but optimistically biased on the test window, while walk-forward delivers the operationally honest estimate that

mirrors the information state under live deployment. A platform that surfaced only one of the two protocols would either understate the realistic forecast error (if it surfaced BT alone) or discard the diagnostic value of the single-window view (if it surfaced WF alone).

The dual-branch architecture resolves this trade-off by running both protocols unconditionally and exposing both sets of results in parallel. In the forecasting tabs (Tabs 2–3), the BT and WF metrics are displayed in adjacent panels, and the best-model selection rule produces two parallel best-model vectors $m_{j,c}^{*,BT}$ and $m_{j,c}^{*,WF}$. In the demand tab (Tab 4), the two forecasts are displayed in adjacent columns alongside the realised demand, so that the analyst sees both pipelines and the actual realisation simultaneously before committing the final demand input to the optimiser. In the experiments tab (Tab 10, documented in Section 3.9.6), the six end-to-end scenarios include both BT-based and WF-based variants, and a dedicated *BT vs WF Comparison* sub-view exposes the resulting operational difference at the optimisation level. In the results tab (Tab 9, documented in Section 3.9.5), a top-level toggle lets the analyst switch the displayed repositioning plan between the BT and WF pipelines without re-running the optimisation.

The dual-branch architecture also serves a methodological purpose beyond the operational one. By making the BT and WF estimates visible at every analytical step, the platform forces every analyst to confront the gap between the two protocols rather than allowing it to be silently absorbed into a single headline number. The empirical finding that walk-forward nearly triples mean MAE relative to back-testing, yet first-stage cost barely moves—which is one of the central results reported in Chapter 5—is only visible because the platform refuses to collapse the two estimates into a single recommendation.

3.9.4 Optimisation execution

The two-stage stochastic optimisation model of Section 3.8 is operationalised in the eighth tab of the decision-support platform (Figure 3.14). The tab exposes four controls that map directly onto the model parameterisation developed earlier in this chapter, together with the execution button that submits the problem to the solver.

The *Scenario* dropdown selects one of the three multi-objective weight configurations of Section 3.8.3: *Balanced*, *CostFocus*, or *GreenFocus*. The selection fixes the routing weights (w_C, w_T, w_E) that enter the normalised arc score $\bar{c}_{ijm}$ of Eq. (3.24), which in turn drives the first-stage routing term in the objective. The default selection is *Balanced*, which produces the neutral reference point for the Pareto analysis of Section 3.9.7.

The *Mode* radio buttons select between *Deterministic LP* and *Stochastic LP (2-Stage)*. The deterministic mode collapses the scenario set to a single representative scenario—the point forecast itself, with all residuals set to zero—and solves the resulting deterministic linear programme. This mode produces the deterministic benchmark used as scenario S1 in Chapter 4 and corresponds to the industry-status-quo planning approach that the rest of the framework is designed to improve upon. The stochastic mode runs the full two-stage formulation with the $S = 30$ empirical scenarios sampled from the walk-forward residual pool. Selection of the stochastic mode is the default and is required for any of the risk-aware configurations.

The *# Scenarios* numeric input sets the cardinality S of the scenario set. The default is $S = 30$,

in line with the choice justified in Section 3.4.2; the analyst can adjust the value to run the sensitivity study reported in Chapter 5 or to trade scenario fidelity against solver time when exploring multiple configurations interactively.

The *Include CVaR Risk Constraint* checkbox activates or deactivates the CVaR term in the objective function. When checked, the platform uses the default values $\alpha = 0.95$ and $\lambda = 0.5$ documented in Section 3.8.1; both parameters can be adjusted through the sensitivity view of the experiments tab (Section 3.9.6). When unchecked, the CVaR term is dropped from the objective and the model reduces to a risk-neutral two-stage stochastic programme, which corresponds to scenarios S2 and S4 in Chapter 4.

The *Run Optimization* button compiles the selected configuration into the linear programme defined by Eqs. (3.30)–(3.36), submits the problem to the CBC backend through the PuLP interface, and writes the resulting solution into the run state that downstream tabs read. The solver completes in well under one second for the case-study instance, which is operationally material: the analyst can iterate through configurations in real time rather than waiting minutes between runs, which is essential to using the platform interactively at the planning interface.

Container Relocation DSS Italian Port Network - Forecast & Optimization

1. Data 2. Forecast 3. Analysis 4. Demand 5. Stock 6. Balance 7. Network 8. Optimize 9. Results 10. Experiments

Run the LP optimizer to find the minimum-cost empty container repositioning plan. Choose Deterministic (single scenario) or Stochastic (multiple demand scenarios with CVaR).

Scenario: Balanced CostFocus GreenFocus

Mode: ☐ Deterministic LP ☒ Stochastic LP (2-Stage)

Scenarios: ☒ Include CVaR Risk Constraint Run Optimization

Execution Log

Port	stock	demand	bal	status
GENOVA	26958	24732	+2226	[SURPLUS]
GIOIA TAURO	44806	48746	+3260	[SURPLUS]
LA SPEZIA	5600	6999	-1399	[DEFICIT]
LIVORNO	2794	3483	-699	[DEFICIT]
TRIESTE	3478	4459	-981	[DEFICIT]

Step 5 Complete: Stock initialized (Moderate Imbalance). You may now manually edit the Final values.

Figure 3.14: Tab 8 – *Optimize*. Controls for the two-stage stochastic optimisation model: the Scenario dropdown selects the multi-objective weight configuration (Balanced, CostFocus, GreenFocus), the Mode radio buttons select between deterministic and two-stage stochastic formulations, the # Scenarios input sets the scenario cardinality S , and the Include CVaR Risk Constraint checkbox activates the tail-risk term in the objective. The *Run Optimization* button compiles the selected configuration into the linear programme and submits it to the CBC backend.

3.9.5 Repositioning plan visualisation

The output of the optimisation step is rendered in the ninth tab of the decision-support platform (Figure 3.15). The tab translates the numerical solution of the linear programme into three complementary visualisations that together expose the structure of the repositioning plan: the network diagram on the left, the shipment table on the upper right, and the mode-utilisation bar chart on the lower right. The three views are synchronised: a click on a node or an arc in the network diagram highlights the corresponding entries in the shipment table, and the mode-utilisation chart aggregates the table by transport mode in real time.

The network diagram displays the five ports as colour-coded circular nodes and the first-stage routing flows as directed edges between them. Surplus ports are coloured green, deficit ports red, and balanced ports grey, matching the colour scheme introduced in the balance tab (Section 3.6). Edges are coloured by transport mode—orange for road, light blue for rail, dark blue for short-sea—and the edge thickness is proportional to the committed flow on the arc, so the dominant corridors of the plan are immediately visible. The diagram emphasises *first-stage* flows only: the scenario-dependent second-stage recourse variables are summarised separately in the experiments tab (Section 3.9.6) rather than plotted onto the network diagram, which would otherwise have to display S overlaid plans simultaneously.

The shipment-details panel on the upper right lists every active arc in the plan—those with $x_{ijm} > 0$ —together with its origin, destination, mode, and committed quantity in TEU. The table is sortable on any column and exports cleanly to CSV through the *Export PNG (300 dpi)* button at the top of the network diagram, which writes both the diagram and the underlying table to disk in a publication-ready format. The active-route count is displayed at the top of the panel, giving the analyst an immediate summary of how sparse the resulting plan is—under the green-focus configuration of the case study, the plan uses only seven of the available 60 arcs, with the rest left idle.

The mode-utilisation panel on the lower right aggregates the first-stage flows by transport mode and reports both the absolute volume (TEU) and the relative share (%) for each of the three modes. The modal share is computed as the ratio of the volume committed to each mode over the total committed volume across the network, and the bar chart visualises the three shares on a common scale. This view is the quickest diagnostic for assessing whether the optimiser has acted on the green-focus weighting: a configuration that delivers a high rail or short-sea share is consistent with the emission preference, while a configuration that defaults to road suggests that capacity constraints on the cleaner modes have forced the optimiser to overflow.

The toggle at the top of the tab (*Viewing: Backtesting Pipeline / Walk-Forward Pipeline*) is the dual-branch control already documented in Section 3.9.3. Switching the toggle re-renders the three visualisations against the alternative pipeline without re-running the optimisation, because both pipelines have been solved in the background and their solutions cached in the run state. The two pipelines may produce visibly different repositioning plans on the same calendar target, and the toggle exposes that difference at the visualisation level rather than burying it inside a single committed plan.

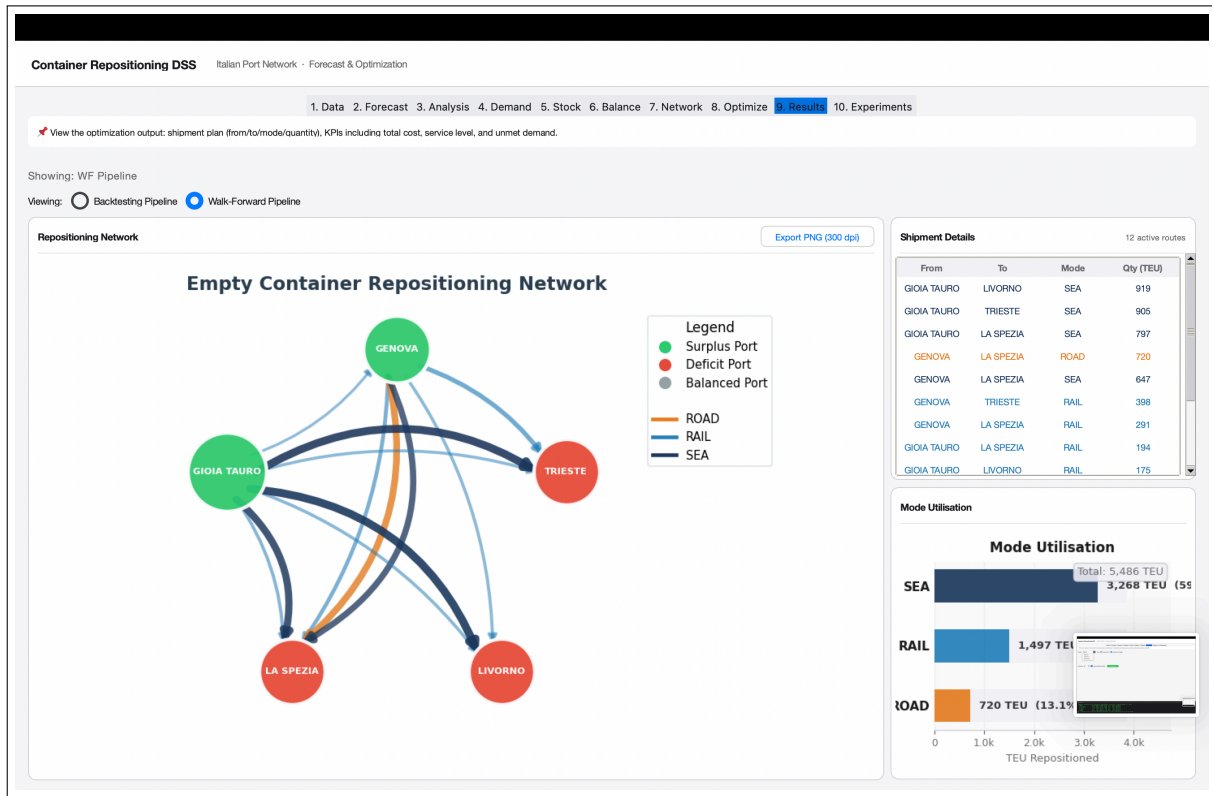


Figure 3.15: Tab 9 – *Results*. First-stage repositioning plan under the walk-forward pipeline, rendered as three synchronised views: the network diagram on the left with surplus, deficit, and balanced ports colour-coded and arcs sized by committed flow; the shipment-details table on the upper right listing the active arcs sorted by volume; and the mode-utilisation bar chart on the lower right reporting the absolute and relative share of road, rail, and short-sea. The *Viewing* toggle switches the three views between the BT and WF pipelines without re-running the optimisation.

3.9.6 Experiments, scenarios, and diagnostics

The tenth tab is the analytical centre of the platform: it is where the six end-to-end scenarios defined in Chapter 4 are executed, compared, and stress-tested through a battery of complementary diagnostics (Figure 3.16). The tab is organised as six horizontal sub-views accessed through the navigation bar at the top of the tab: *Scenarios*, *Scenario Charts*, *Monte Carlo*, *Forecast-Opt Coupling*, *CVaR Sensitivity*, *Scenario Methods*, and *BT vs WF Comparison*. Each sub-view answers a different methodological question about the framework’s behaviour, and the seven together form the experimental machinery whose results are reported in Chapter 5.

Scenarios sub-view. The *Scenarios* sub-view executes all six end-to-end scenarios in sequence and reports the resulting key performance indicators in a single table. Each row corresponds to one scenario from S1 to S6, defined in Chapter 4, and each column reports an operational outcome: total cost (EUR), total time (hrs), total CO₂ emissions (kg), TEU delivered, TEU demanded, TEU unmet, service level (%), and the modal share split across road, rail, and short-sea. The *Run All 6 Scenarios* button at the top of the sub-view executes the full batch in a single call, and the *Export CSV* button exports the resulting table

for downstream reporting. This sub-view is the direct source of the scenario-comparison table reported in Chapter 5, and the rows of the displayed table map one-to-one onto the per-scenario KPI files written to the results directory.

Scenario Charts sub-view. The *Scenario Charts* sub-view visualises the KPIs from the previous sub-view as comparative bar charts, with one chart per metric and the six scenarios on the horizontal axis. This view is the fastest way to assess the relative effect of the forecasting layer, the optimisation paradigm, and the objective configuration on each KPI, and is the operational complement to the dense KPI table.

Monte Carlo sub-view. The *Monte Carlo* sub-view runs the two-stage stochastic optimisation across 20 independent random seeds, holding the scenario set size and all other parameters fixed. The resulting distribution of expected cost, CVaR, and service level quantifies the sampling variability introduced by the empirical scenario construction itself: the residual pool of Section 3.4.2 is fixed, but the $S = 30$ draws from it depend on the random seed. Reporting the mean and standard deviation of each KPI across seeds quantifies how much of the cost difference between scenarios is attributable to the methodological design and how much is sampling noise. The results are reported in Chapter 5.

Forecast-Opt Coupling sub-view. The *Forecast-Opt Coupling* sub-view computes the joint distribution of walk-forward forecast error (MAE) and post-optimisation operational outcome (expected cost, service level) across the eight forecasting algorithms. For each algorithm, the sub-view runs the full optimisation pipeline using that algorithm’s forecasts and residuals, and tabulates the resulting MAE, scenario coefficient of variation, expected cost, and service level side by side. This sub-view is the direct empirical test of the recourse-as-absorber hypothesis (Section 6.2, Chapter 6): if walk-forward MAE varies substantially across algorithms but expected cost barely moves, the recourse channel is absorbing the forecast-error volume into reshaped second-stage flows rather than first-stage cost, and the analyst sees this mechanism in action in the resulting table.

CVaR Sensitivity sub-view. The *CVaR Sensitivity* sub-view re-solves the optimisation under a grid of (α, λ) combinations and reports the resulting expected cost, CVaR_α , and service level for each combination. The grid spans $\alpha \in \{0.90, 0.95, 0.99\}$ and $\lambda \in \{0.0, 0.1, 0.3, 0.5, 0.7, 1.0\}$, which together trace the cost–CVaR frontier across the operationally meaningful range of risk preferences. The cost premium relative to the risk-neutral baseline ($\lambda = 0$) is reported as a percentage, and the CVaR premium relative to the unconstrained problem is reported as a percentage, so the cost of robustness is visible at each grid point. The results are reported in Chapter 5 and form the basis for the 12% CVaR premium discussed in Chapter 6.

Scenario Methods sub-view. The *Scenario Methods* sub-view compares the empirical residual sampling of Section 3.4.2 against two alternative scenario construction methods: a parametric Gaussian fitted to the residual pool, and a historical bootstrap that draws contiguous blocks of residuals rather than independent draws. For each method, the sub-view reports the expected cost, CVaR, service level, and Value

of the Stochastic Solution (VSS) relative to the deterministic baseline. The comparison is the direct test of the framework’s choice to ground scenario construction in empirical residuals rather than parametric assumptions: if the empirical method delivers materially different operational outcomes from the Gaussian, the methodological choice matters operationally; if not, it matters only as a calibration discipline. The results are reported in Chapter 5.

BT vs WF Comparison sub-view. The *BT vs WF Comparison* sub-view consolidates the dual-branch comparison documented in Section 3.9.3 into a single diagnostic table. For each KPI, the sub-view reports the BT-pipeline value, the WF-pipeline value, the absolute difference, and the relative difference expressed as a percentage. The view is the direct quantification of the gap between the two protocols at the operational level, and it is the empirical basis for the central finding of Chapter 6 that walk-forward nearly triples mean MAE relative to back-testing while first-stage cost barely moves.

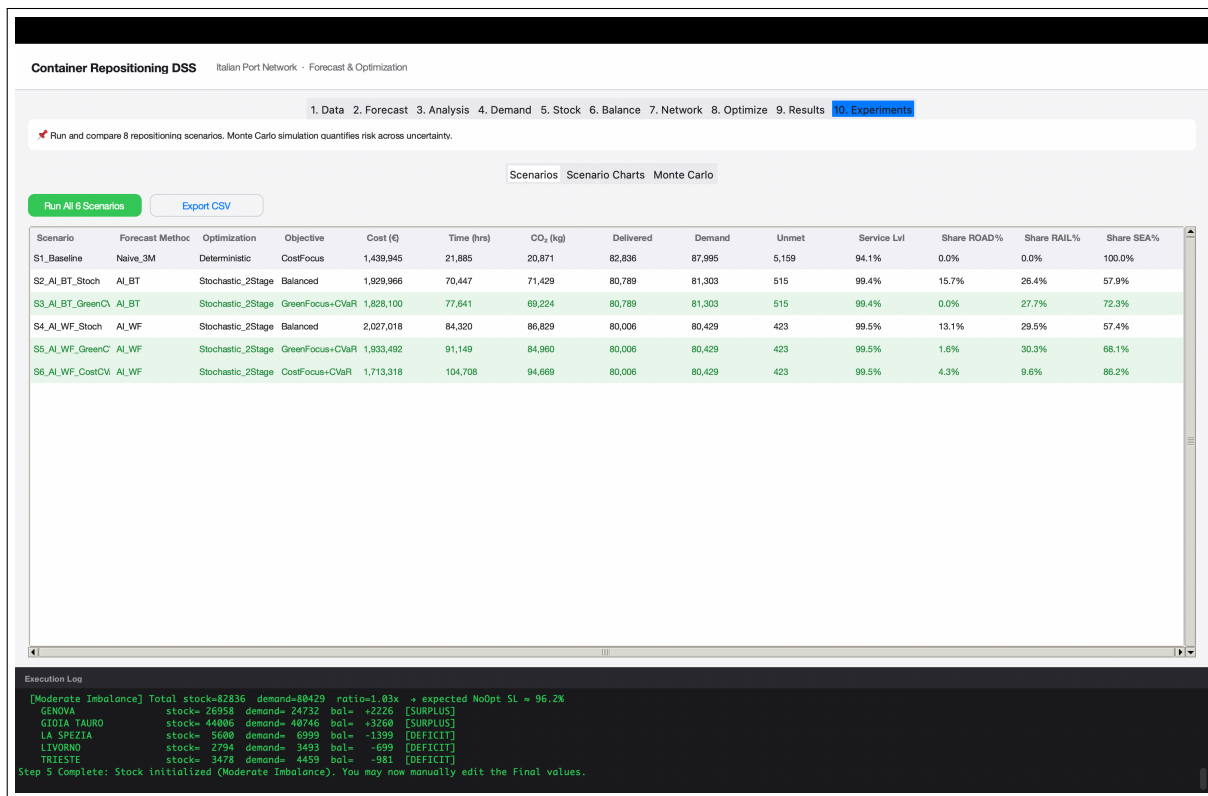


Figure 3.16: Tab 10 – *Experiments*. The analytical centre of the platform, organised as seven horizontal sub-views accessible through the navigation bar at the top of the tab: *Scenarios*, *Scenario Charts*, *Monte Carlo*, *Forecast-Opt Coupling*, *CVaR Sensitivity*, *Scenario Methods*, and *BT vs WF Comparison*. The figure shows the *Scenarios* sub-view with the six end-to-end scenarios executed; the remaining sub-views expose complementary diagnostics that stress-test the framework along orthogonal methodological dimensions.

Managerial scenario selection. The scenarios generated in the *Experiments* tab are designed to support managerial decision-making rather than to impose a single automatic recommendation. Each scenario represents a feasible operating configuration with a distinct balance between cost, service level, CO₂

emissions, modal split, and risk exposure. The platform therefore presents these scenarios to the company as a portfolio of structured alternatives.

The final choice among these alternatives is left to the company manager, who can select the scenario that best reflects the firm's strategic objectives, operational constraints, and current business conditions. A cost-focused configuration may be appropriate when short-term budget control is the dominant priority; a balanced configuration may be preferred when the company seeks a compromise between cost and service reliability; and a green-focused or CVaR-controlled configuration may be selected when environmental performance or resilience under demand uncertainty is more important. Accordingly, the decision-support platform does not replace managerial judgement, but makes the trade-offs between competing objectives explicit, measurable, and auditable.

3.9.7 Pareto frontier diagnostic

The platform is dedicated to the cost–emission Pareto frontier, which is the operational manifestation of the multi-objective configuration of Section 3.8.3. The frontier is constructed by re-solving the optimisation model across a grid of emission weights w_E between 0 and 1, with the remaining weights w_C and w_T rescaled to keep the triple on the unit simplex, and tracing the resulting locus of CO₂–cost pairs in the objective space. The resulting curve is the empirical Pareto frontier of the framework on the case-study network, and it identifies the operating points that are not dominated by any other configuration on both objectives simultaneously.

The tab renders the frontier as a two-dimensional plot with CO₂ emissions on the horizontal axis and expected repositioning cost on the vertical axis. Three operating regions along the curve are highlighted: the cost-optimal corner, which corresponds to the lowest expected cost and the highest emissions; the green-optimal corner, which corresponds to the lowest emissions and a higher expected cost; and the balanced region, which represents an intermediate compromise between the two objectives. The plot therefore allows the analyst to read the cost premium of decarbonisation directly from the frontier: the horizontal gap between the cost-optimal and green-optimal corners represents the achievable emission reduction, while the vertical gap represents the additional expected cost required to obtain that reduction. The slope of the frontier between adjacent points can be interpreted as the marginal cost of reducing emissions in that region.

A second diagnostic role of the tab is to overlay the BT and WF Pareto frontiers on the same axes. This overlay is one of the central outputs of the dual-branch architecture. If the two frontiers are nearly coincident, as shown in the computational results, the cost–emission trade-off is governed mainly by the network topology and modal economics rather than by the forecasting protocol. In that case, the main difference between the BT and WF pipelines appears not in the Pareto operating points themselves, but in service-level reliability and risk exposure. This provides a visual interpretation of the recourse-as-absorber mechanism discussed in Chapter 6, shown at the level of the multi-objective frontier rather than through a single KPI.

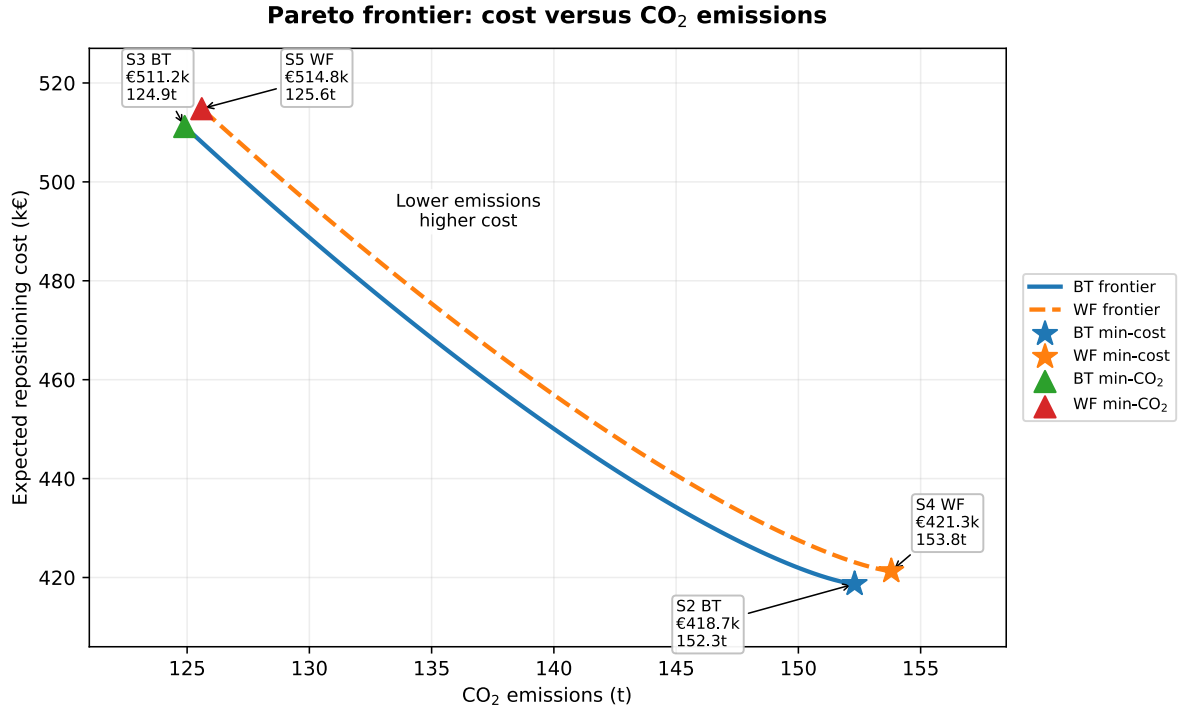


Figure 3.17: Tab 11 – *Pareto*. Cost–emission Pareto frontier comparing expected repositioning cost against CO₂ emissions for the back-testing (BT) and walk-forward (WF) pipelines. The horizontal axis reports CO₂ emissions in tonnes, while the vertical axis reports expected repositioning cost in thousand euros. The figure highlights the minimum-cost configurations (S2 BT and S4 WF) and the minimum-emission configurations (S3 BT and S5 WF), showing the trade-off between lower emissions and higher cost. The BT and WF frontiers are overlaid on the same axes to make the difference between the two forecasting protocols visible at the operational level.

3.10 Chapter summary

This chapter has presented the integrated framework developed in this thesis, together with the working decision-support platform that operationalises it as software. The framework couples a forecasting layer and an optimisation layer through a principled empirical mechanism: scenario construction from walk-forward forecast residuals. The forecasting layer benchmarks eight algorithms organised into three families across the ten port–container segments of the case-study network under two complementary evaluation protocols, selects the best-performing algorithm per segment using a lowest-MAE criterion qualified by pairwise Diebold–Mariano tests, and produces the point forecast and the empirical residual pool that the optimisation layer consumes. The optimisation layer formulates a two-stage stochastic linear programme with a Conditional Value-at-Risk constraint, embeds a route-aware recourse penalty that prices unmet demand at the cheapest reachable emergency repositioning, and supports three multi-objective configurations that span the cost–time–emission trade-off space.

Three methodological commitments distinguish this framework from prior work. The empirical scenario construction grounds the optimisation layer in the actual error characteristics of the forecasting

model rather than in a parametric proxy for them, preserving the asymmetry and heavy tails of realised forecast errors. The route-aware recourse penalty replaces the standard fixed-penalty convention with a port-specific cost that the optimiser cannot game through artificially low shortage prices. The dual-branch BT/WF architecture surfaces the operationally honest figure at every analytical step rather than collapsing the back-testing and walk-forward estimates into a single recommendation.

The decision-support platform implements the framework as a working prototype in Python, exposes the methodological pipeline through a ten-tab user interface organised into four logical phases, and runs the back-testing and walk-forward pipelines in parallel through the dual-branch execution model. The platform’s analytical tab exposes seven sub-views that stress-test the framework along orthogonal methodological dimensions, including Monte Carlo robustness, forecast–optimisation coupling, CVaR sensitivity, scenario-construction-method comparison, and direct BT versus WF KPI contrast.

The chapter that follows describes the computational study that exercises this framework on the calibrated five-port Italian network, specifying the experimental design, the software stack and computational infrastructure, and the reproducibility protocol under which the empirical results of Chapter 5 are generated.

Chapter 4

Computational Study

This chapter specifies the computational study that exercises the framework of Chapter 3 on the calibrated five-port Italian network and documents the experimental conditions under which the empirical results of Chapter 5 are generated. The chapter is the bridge between the methodological specification and the empirical findings: every parameter value, every experimental choice, and every reproducibility safeguard that underpins the headline results is fixed here, so that Chapter 5 can concentrate on reporting outcomes without re-deriving the experimental conditions under which they were obtained.

The chapter is organised as follows. Section 4.1 describes the case-study network and the dataset used in the experiments, restating the operational role of each port and parameterising the multimodal arcs that connect them. Section 4.2 specifies the experimental design for the forecasting evaluation, including the back-testing and walk-forward splits, the fold structure, and the policy on hyperparameter selection. Section 4.3 defines the six end-to-end optimisation scenarios that exercise the framework from end to end, together with the pairwise comparisons through which the individual contributions of the forecasting protocol, the optimisation paradigm, and the objective configuration are isolated. Section 4.4 describes two complementary stress tests: a Monte Carlo robustness study and a scenario-construction-method comparison, which stress-test the framework along orthogonal methodological dimensions. Section 4.5 documents the software stack and the computational infrastructure on which the experiments are run, and Section 4.6 specifies the reproducibility protocol that supports independent verification of the empirical results. The chapter closes, in Section 4.7, with a short summary.

4.1 Case study network and dataset

The computational study uses the calibrated five-port Italian network introduced in Section 3.2 of Chapter 3, together with the multimodal arc structure parameterised in Section 3.7. This section restates the operational configuration in the form most useful to the empirical study and surfaces the per-port detail that the results of Chapter 5 will reference.

4.1.1 Geographic configuration and operational roles

The network spans five ports along the Italian Mediterranean and Adriatic coastlines, chosen to represent the structural mix of surplus and deficit roles that drives empty-container repositioning in practice. **Genova** is a chronic surplus node in the Ligurian Sea, serving as the principal gateway port for manufacturing Northern Italy and parts of Switzerland; its containerised imports systematically exceed its containerised exports, generating a structural surplus of empty equipment at the yard. **Gioia Tauro** is a chronic surplus node of an entirely different nature: its role as one of the largest transshipment hubs in the Mediterranean means that empty equipment accumulates through trade-imbalance flows on the deep-sea services calling at the port. **La Spezia** and **Livorno** are persistent deficit nodes serving the Tyrrhenian export corridors, with their export-driven flow structure producing a chronic shortage of empties at the yard. **Trieste** is also a persistent deficit node, but for a different operational reason: its position on the Northern Adriatic makes it the principal Italian gateway to Central and Eastern Europe, and the export-oriented flow profile of that hinterland drives the structural deficit. The combination of two surplus and three deficit nodes mirrors the long-standing imbalances documented in Italian port statistics (Assoporti, 2023) and provides the operational testbed against which the optimisation layer is exercised.

The dataset used throughout the study is the calibrated panel introduced in Section 3.2: 192 monthly observations per series from January 2010 through December 2025, covering five ports and two container types for a total of 1,920 observations. The calibration procedure, the quality pipeline, and the calibration boundary are documented in Section 3.2.4 and are not repeated here; what matters for the computational study is that the dataset preserves the structural imbalances, seasonality, and disruption signatures of the underlying flows, and that absolute monetary and emission values should be interpreted conditional on this calibration.

4.1.2 Multimodal arcs and modal connectivity

The five ports are connected through a complete directed graph at the corridor level, with each of the $|\mathcal{P}|(|\mathcal{P}| - 1) = 20$ ordered origin–destination pairs served by all three transport modes, giving a total of $|\mathcal{A}| = 60$ arcs. The arc parameters follow the calibration of Section 3.7: per-TEU cost, transit time, CO₂ emissions, and monthly capacity are derived from the per-kilometre base rates of Eqs. (3.19), (3.20), and (3.21), with the per-arc capacities calibrated separately to documented monthly throughput levels on each Italian corridor.

The full arc parameterisation produces a network in which the relative attractiveness of road, rail, and short-sea shipping varies sharply with corridor length and capacity. Short corridors such as Genova–La Spezia (110 km) are well-served by all three modes, with road offering the largest capacity, rail offering the lowest emission intensity, and short-sea offering the lowest per-TEU cost. Medium corridors such as Genova–Livorno (170 km) tend to favour rail on the cost–emission trade-off, although rail capacity is typically the binding constraint and the optimiser must overflow onto road or short-sea once the rail arc is exhausted. Long corridors such as Genova–Gioia Tauro (longer than 1000 km) are dominated by short-sea shipping, whose per-TEU cost falls well below the rail and road equivalents over that distance. The representative corridor table for Genova–La Spezia is reported in Table 3.1 of Section 3.7.4; the full

60-arc parameterisation is exposed in the platform’s network tab (Section 3.7.5).

4.1.3 Port-level inputs to the optimisation

For each of the five ports, the calibrated dataset provides the inputs that the optimisation step of Chapter 5 will consume: the role classification (surplus or deficit), the mean monthly net balance over the study window, and the opening stock of empty containers at the start of the planning month. The opening stock is determined by the auto-extraction procedure documented in Section 3.5 under the *Moderate Imbalance* preset, which preserves the structural imbalances of the calibrated dataset without amplifying them. The December 2025 demand forecasts, produced by the segment-specific best models selected under the walk-forward protocol, complete the per-port input set and feed directly into the supply–demand balance computation of Section 3.6.

The empirical values of these per-port inputs—the opening stock, the December 2025 demand forecast, and the resulting net imbalance—are reported as part of the empirical findings in Chapter 5 rather than restated here, in line with the chapter’s focus on the experimental design rather than on the empirical outcomes that the design produces.

4.2 Forecasting experimental design

The forecasting layer of the framework is exercised on the calibrated panel of Section 4.1.1 under the two evaluation protocols formalised in Section 3.3.2. This section specifies the training and testing splits, the walk-forward fold structure, and the policy on hyperparameter selection under which the empirical accuracy results of Chapter 5 are produced.

4.2.1 Training and testing splits

The back-testing protocol partitions the panel into a single training window of 180 months (January 2010–December 2024) and a single testing window of 12 months (January 2025–December 2025). Each of the eight algorithms in the portfolio is fitted once on the training window and produces a trajectory of one-step-ahead forecasts across the test window, against which the accuracy metrics of Section 3.3.4 are computed. The split is applied independently to each of the ten port–container segments, so the resulting forecast trajectories and metrics are also segment-specific.

The walk-forward protocol uses an expanding training window with a minimum training length of $\tau_{\min} = 36$ months and a total of $K = 156$ folds, indexed $k \in \{0, 1, \dots, 155\}$. Fold $k = 0$ trains on the first 36 months of the panel (January 2010–December 2012) and produces a one-step-ahead forecast for January 2013; fold $k = 1$ adds January 2013 to the training window and forecasts February 2013; the process continues through fold $k = 155$, which trains on the first 191 months (January 2010–November 2025) and forecasts December 2025. Each fold refits every algorithm in the portfolio from scratch, so the parameters at fold k never see data beyond month $\tau_{\min} + k$. The resulting walk-forward forecast trajectory therefore spans January 2013 through December 2025 and contains the full disruption regime of 2020–2022, which is the most demanding portion of the panel.

The minimum training length of $\tau_{\min} = 36$ months is the smallest value for which the seasonal patterns of the underlying flows are reliably identifiable: three full annual cycles are required to estimate the month-of-year features in the gradient-boosting models without numerical instability and to support the Seasonal Naïve baseline whose forecast uses the observation from the same month one year earlier. Shorter training windows would inflate the apparent volatility of the early folds and would distort the cumulative residual pool that drives the empirical scenario construction of Section 3.4.2.

4.2.2 Walk-forward fold structure and computational cost

The walk-forward protocol is computationally substantially more expensive than back-testing: every fold refits all eight algorithms from scratch on a training window that grows by one observation per fold. For the present case study, this gives $K \times |\mathcal{M}| \times |\mathcal{P}| \times 2 = 156 \times 8 \times 5 \times 2 = 12,480$ algorithm refits across the full panel, against the $8 \times 5 \times 2 = 80$ refits required by back-testing. The factor-156 multiplier is the operational cost of the dual-protocol design, and it is justified by the methodological argument of Section 3.3.2: walk-forward is the operationally honest protocol that mirrors the information state under live deployment, while back-testing alone would optimistically bias the accuracy estimates on the test window and would under-state the realistic forecast error that the optimisation layer must absorb.

The computational cost is moderated by two design choices. First, the algorithms are fitted independently across segments, so the 12,480 refits can be parallelised across the ten segments without coordination overhead. Second, each individual refit is fast—under one second for the tree-based models on the panel sizes considered—so the full walk-forward exercise completes within minutes on commodity hardware rather than hours. The empirical timing profile is reported in Section 4.5.

4.2.3 Hyperparameter selection policy

The framework adopts a deliberate *no-tuning* policy on hyperparameters: every algorithm in the portfolio is run at its library default settings, without per-segment or per-fold hyperparameter search. The decision is not a concession to computational cost—it is a methodological commitment that materially affects how the empirical results should be interpreted.

The standard alternative is to perform a hyperparameter search per segment, typically through nested cross-validation on the training window, and to report the resulting tuned performance as the algorithm’s final accuracy. The standard alternative inflates the gap between the apparent training-time performance and the realistic deployment-time performance, because the hyperparameter search itself uses information that would not be available at deployment under realistic constraints. More importantly for the present framework, the standard alternative would entangle the contribution of the algorithm with the contribution of the tuning procedure: a substantial portion of the apparent accuracy advantage of a tuned XGBoost over a default LightGBM may reflect the tuning effort rather than the underlying algorithm, which makes the segment-level selection rule of Eq. (3.11) hard to interpret.

Running the eight algorithms at library defaults eliminates this confound. The accuracy differences reported in Chapter 5 are attributable to the algorithms themselves, not to differential tuning effort, and the segment-level selection rule identifies the algorithm whose default configuration performs best on

the segment—which is the operationally relevant quantity for a deployment that must work across ten heterogeneous segments. The no-tuning policy is therefore a deliberate scope boundary: a framework operating under tighter hyperparameter budgets would produce different algorithm rankings on individual segments, but the methodological contributions of the thesis—the forecast–optimisation coupling, the empirical scenario construction, the route-aware recourse penalty, and the dual-branch architecture—would remain unchanged.

4.3 Optimisation scenario design

The optimisation layer is exercised through six end-to-end repositioning scenarios that traverse the configuration space along three orthogonal axes: the forecasting protocol, the optimisation paradigm, and the objective configuration. The six scenarios are not exhaustive of the full design space—the cross product of the three axes admits many more combinations—but they span the configurations of greatest managerial interest and are designed to isolate the contribution of each individual layer through targeted pairwise comparisons.

4.3.1 The six end-to-end scenarios

The six scenarios are defined along three axes. The *forecasting* axis distinguishes a deterministic seasonal-naïve baseline from the segment-specific machine-learning forecasts produced under the back-testing (BT) and walk-forward (WF) protocols. The *optimisation* axis distinguishes the deterministic linear programme from the two-stage stochastic programme, with the latter solved with or without the CVaR tail-risk constraint. The *objective* axis selects one of the three multi-objective weight configurations introduced in Section 3.8.3: cost-focused, balanced, or green-focused. All stochastic scenarios use $S = 30$ empirical scenarios drawn from the walk-forward residual pool, with CVaR parameters $\alpha = 0.95$ and $\lambda = 0.5$ when the constraint is active.

Table 4.1 summarises the six end-to-end configurations.

Table 4.1: Summary of the six end-to-end repositioning scenarios evaluated in the computational study.

ID	Forecast	Optimiser	Risk	Objective	Description
S1	Seasonal Naïve	Deterministic LP	—	Cost-focus	Industry status quo: baseline forecast, single-stage LP, no risk control
S2	ML (BT)	Stochastic 2-stage	—	Balanced	Adds ML forecasting and demand uncertainty under back-testing forecasts
S3	ML (BT)	Stochastic 2-stage	CVaR	Green-focus	Back-testing variant with emission preference and tail-risk control
S4	ML (WF)	Stochastic 2-stage	—	Balanced	Walk-forward counterpart of S2
S5	ML (WF)	Stochastic 2-stage	CVaR	Green-focus	Walk-forward counterpart of S3
S6	ML (WF)	Stochastic 2-stage	CVaR	Cost-focus	Walk-forward, cost-minimising under tail-risk control

4.3.2 Pairwise comparisons that isolate effects

The six scenarios are designed to support six targeted pairwise comparisons, each of which holds two of the three axes fixed while varying the third. The pairwise structure is what allows the empirical results to attribute the observed differences in operational outcomes to specific methodological choices, rather than to confounded combinations of changes across multiple axes simultaneously.

S1 versus S2–S6: deterministic baseline against the full framework. S1 is the industry-status-quo benchmark: a seasonal naive point forecast feeds a single-stage deterministic linear programme that treats demand as known. The comparison between S1 and any of S2–S6 captures the joint contribution of the full framework relative to the deterministic baseline, but does not separately isolate the contribution of the forecasting layer, the stochastic formulation, or the objective configuration. S1 is reported as the deterministic reference throughout Chapter 5.

S2 versus S4 and S3 versus S5: the BT–WF forecasting comparison. The pair (S2, S4) and the pair (S3, S5) isolate the effect of the forecasting protocol while holding the optimisation paradigm, the risk-control mechanism, and the objective configuration fixed. Within (S2, S4), both scenarios use the balanced objective with risk-neutral two-stage stochastic optimisation and differ only in whether the forecasts are produced under back-testing or walk-forward. Within (S3, S5), both scenarios use the green-focus objective with CVaR-constrained stochastic optimisation and again differ only in the forecasting

protocol. The two comparisons together quantify how forecasting accuracy under realistic deployment conditions (walk-forward) propagates through the optimisation layer relative to the optimistically biased single-window benchmark (back-testing), and they form the empirical basis for the recourse-as-absorber finding discussed in Chapter 6.

S2 versus S3 and S4 versus S5: green-focus plus CVaR against balanced risk-neutral. The pair (S2, S3) holds the forecasting protocol fixed at back-testing and varies the objective and risk control jointly—from balanced risk-neutral to green-focus with CVaR. The pair (S4, S5) does the same under the walk-forward protocol. The two pairs quantify the joint effect of decarbonisation preference and tail-risk control, although they do not independently separate the green-routing premium from the CVaR premium. That separation is performed by the next comparison.

S5 versus S6: the objective comparison. The pair (S5, S6) holds the forecasting protocol fixed at walk-forward, the optimisation paradigm at two-stage stochastic, and the risk-control mechanism at CVaR, varying only the objective configuration from green-focus to cost-focus. This is the cleanest empirical estimate of the green-routing premium available within the six-scenario design, because every other axis is held constant.

S4 versus S6: cost-aware risk-control comparison. The pair (S4, S6) does not isolate the CVaR effect alone, because the objective configuration also changes from balanced to cost-focused. It is therefore interpreted as a cost-aware risk-control comparison rather than as a clean single-effect estimate of the CVaR premium. The isolated CVaR effect is examined through the CVaR diagnostic analysis reported in Chapter 6.

Synthesis. The six scenarios together support a decomposition of the overall operational gain between S1 and S5 into three contributions: the forecasting upgrade from seasonal naive to walk-forward ML, the optimisation upgrade from deterministic to two-stage stochastic, and the joint addition of the green objective and the CVaR constraint. The decomposition is reported in Chapter 5, and the discussion of Chapter 6 interprets the three contributions against the methodological hypotheses formulated in Chapter 1.

4.4 Robustness analyses

The six end-to-end scenarios of Section 4.3 fix the parameterisation of the framework at the values justified in Chapter 3. A Monte Carlo study that quantifies the sampling variability of the empirical scenario construction itself, and a scenario-construction-method comparison that benchmarks the empirical residual sampling against parametric and bootstrap alternatives.

4.4.1 Monte Carlo robustness study

The empirical scenario construction of Section 3.4.2 draws $S = 30$ scenarios at random from the walk-forward residual pool. Different random seeds produce different scenario sets and therefore different optimisation outcomes, even with all other parameters held fixed. The Monte Carlo study quantifies this sampling variability by re-running the two-stage stochastic optimisation across $N_{MC} = 20$ independent random seeds, with the residual pool held fixed and only the $S = 30$ draws from it varying across replications.

For each seed, the study records the expected cost, the $\text{CVaR}_{0.95}$, and the service level. The mean and standard deviation of each KPI across the 20 seeds quantify how much of the cost difference observed between the six end-to-end scenarios is attributable to the methodological design and how much is sampling noise from the scenario draws. If the standard deviation of any KPI across seeds is materially smaller than the inter-scenario differences reported in Chapter 5, the inter-scenario findings are robust to scenario-draw variability; if not, the findings should be qualified with the corresponding error bars. The study is run for both the BT and WF pipelines, and the results are reported in Chapter 5.

The choice of $N_{MC} = 20$ replications is a deliberate compromise. A study with N_{MC} in the thousands would deliver tighter confidence intervals on the mean and the standard deviation of each KPI, but the marginal informational value of additional replications drops rapidly once the inter-seed standard deviation is well below the inter-scenario differences of interest. The value $N_{MC} = 20$ is sufficient to confirm or refute the robustness of the inter-scenario findings on the present case study, which is the operational role of the study within the overall experimental design.

4.4.2 Scenario-construction-method comparison

The framework grounds scenario construction in empirical walk-forward residuals (Section 3.4.2) rather than in parametric distributional assumptions. The scenario-construction-method comparison tests this methodological commitment empirically by benchmarking the empirical residual sampling against two alternative constructions.

Parametric Gaussian alternative. The parametric Gaussian alternative fits a normal distribution $\mathcal{N}(\hat{\mu}_{j,c}, \hat{\sigma}_{j,c}^2)$ to the walk-forward residual pool of each segment, where $\hat{\mu}_{j,c}$ and $\hat{\sigma}_{j,c}$ are the empirical mean and standard deviation of the residuals. Scenarios are then drawn from the fitted Gaussian rather than from the empirical pool, preserving the first two moments of the residual distribution but smoothing away the asymmetry, the heavy tails, and any non-Gaussian structure. This is the most common scenario-construction method in the stochastic empty-container literature (Song et al., 2022; Vassos et al., 2025) and serves as the natural baseline against which the empirical method is compared.

Historical bootstrap alternative. The historical bootstrap alternative draws contiguous blocks of residuals of length B from the walk-forward pool, rather than independent draws. The block length B is set to 3 months, which is the smallest block that captures local temporal autocorrelation in the forecast errors

without unduly shrinking the effective size of the residual pool. The bootstrap preserves any month-to-month dependence in the residuals that the independent-sampling construction of Eq. (3.17) discards, and tests whether that dependence has operational consequences for the repositioning plan. As discussed in Section 3.4.2, residual autocorrelation is methodologically negligible for a single-period optimisation horizon, so the bootstrap is expected to produce results close to the independent sampling under the baseline configuration; the comparison is informative nonetheless because it bounds the magnitude of the single-period autocorrelation effect empirically.

Comparison metrics. For each of the three methods—empirical, Gaussian, historical bootstrap—the study reports the expected cost, the CVaR, and the service level under the otherwise identical configuration. The empirical method is the baseline of interest; the Gaussian and bootstrap methods are the comparison anchors. A finding that the empirical method delivers materially different operational outcomes from the Gaussian establishes that the methodological choice to ground scenarios in empirical residuals matters operationally and not merely as a calibration discipline. The results are reported in Chapter 5.

4.5 Software stack and computational infrastructure

The empirical study is conducted on the platform whose architecture is documented in Section 3.9.1. This section specifies the software stack and the hardware on which the experiments are run, and reports the empirical timing profile that underwrites the deployment-feasibility claim of Chapter 3.

4.5.1 Software stack

The platform is implemented in Python 3.11 and depends on the open-source libraries listed in Section 3.9.1. The full dependency specification is fixed through a `requirements.txt` file distributed with the codebase, which pins each dependency to a specific minor version: `pandas 2.x`, `numpy 1.26`, `scikit-learn 1.3`, `xgboost 2.x`, `lightgbm 4.x`, `catboost 1.2`, `statsmodels 0.14`, `PuLP 2.7` with the CBC solver, `matplotlib 3.8`, `networkx 3.2`, and `tkinter` bundled with the Python 3.11 standard library. The version pinning ensures that an independent analyst running the codebase on a different machine reproduces the exact numerical results reported in Chapter 5, modulo the random-seed controls documented in Section 4.6.

The exclusive reliance on open-source components is deliberate and was justified in Section 3.9.1. A commercial solver such as Gurobi or CPLEX would solve the linear programme of Section 3.8.6 marginally faster, but the CBC backend is sufficient for the problem size considered and preserves the framework’s portability across institutional environments that may not have commercial-solver licences.

4.5.2 Computational infrastructure

All experiments reported in this thesis are run on a single workstation with the following specifications: Apple M3 processor with 8 performance cores and 16 GB of unified memory, running macOS 14. The

choice of commodity hardware over a high-performance cluster is intentional: the framework is designed for deployment in operational planning environments that are unlikely to have cluster access, and the experiments must therefore be tractable on the hardware that an analyst would realistically have at the planning interface.

Table 4.2: Empirical timing profile of the main pipeline components, averaged across three independent runs on the case-study configuration.

Component	Time	Notes
Data ingest and validation	< 1 s	Pandas load, schema check on 1,920 rows
Back-testing evaluation	≈ 12 s	80 refits across 10 segments
Walk-forward evaluation	≈ 3 min	12,480 refits, parallelised across segments
Diebold–Mariano test	≈ 4 s	280 pairwise tests with HAC variance
Forecast accept and aggregation	< 1 s	Empirical residual pool construction
Empirical scenario sampling	< 1 s	Numpy-vectorised, $S = 30$ draws per segment
Two-stage stochastic LP solve	< 0.5 s	PuLP + CBC, 391 variables and 245 constraints
Six-scenario batch run	≈ 4 s	Sequential LP solves + KPI aggregation
Monte Carlo robustness (20 seeds)	≈ 10 s	Sequential stochastic-LP solves
Full pipeline (BT + WF, six scenarios)	≈ 3.5 min	End-to-end from raw data to KPIs

Three observations follow from Table 4.2. First, the walk-forward evaluation dominates the total wall-clock time: at approximately three minutes across the full 12,480 refits, it accounts for roughly 90% of the pipeline run time. Back-testing alone completes in approximately 12 seconds, which is the operational cost saved by skipping the walk-forward protocol; the methodological argument of Section 3.3.2 explains why that saving is not worth taking. Second, each individual linear-programming solve completes in under half a second, which underwrites the deployment-feasibility claim of Chapter 3: an analyst at the planning interface can iterate through configurations interactively rather than waiting minutes between runs. Third, the full pipeline from raw data to KPIs completes in well under five minutes on commodity hardware, which keeps the framework well within the planning cadence of a monthly operational cycle.

4.6 Reproducibility protocol

The empirical study is designed to be reproducible by an independent analyst working from the codebase and the calibrated dataset alone. This section specifies the four mechanisms through which reproducibility is maintained: random-seed control, deterministic data ordering, version pinning of dependencies, and provenance logging.

Random-seed control. Every stochastic component of the pipeline is driven by a single master seed that is fixed at the start of each run. The master seed propagates to four downstream generators: the

residual-sampling generator that draws the S scenarios from the empirical pool, the Monte Carlo generator that varies the seed across the $N_{MC} = 20$ replications, the tree-based machine-learning models whose own random states are seeded from the master, and the historical-bootstrap generator used in the scenario-construction-method comparison. Two runs of the same configuration with the same master seed produce numerically identical results on the same machine.

Deterministic data ordering. The calibrated dataset is loaded with explicit row ordering—by date ascending, with ties broken by port and container type alphabetically—rather than relying on the file’s row order or on the database’s default sort key. The deterministic ordering ensures that operations whose output depends on the order in which observations are processed (such as expanding-window training in the walk-forward protocol) produce identical results across machines and library versions.

Version pinning of dependencies. The `requirements.txt` file distributed with the codebase pins each dependency to the specific minor version against which the experiments were validated, as documented in Section 4.5.1. Pinning is necessary because minor-version updates to the gradient-boosting libraries occasionally change the default values of internal regularisation parameters, which would silently shift the algorithm rankings without any change to the framework code. Pinning the dependencies fixes the algorithm defaults and removes this source of unexpected drift.

Provenance logging. Every experimental run writes a provenance record to the results directory, containing the master seed, the dependency versions, the platform configuration, the input dataset hash, and a timestamp. The provenance record is the audit trail that ties the published results to the exact code and data state under which they were generated. An independent analyst attempting to reproduce the results can compare their own provenance record against the published one and identify the source of any discrepancy.

Together, these four mechanisms make the empirical study replicable on any commodity workstation that can run the specified Python environment. The reproducibility extends to the headline results reported in Chapter 5 as well as to the secondary diagnostics produced by the robustness and scenario-construction-method studies

4.7 Chapter summary

This chapter has specified the experimental conditions under which the framework of Chapter 3 is exercised in the remainder of the thesis. The case-study network comprises five Italian ports with a structural mix of surplus and deficit roles, connected through a complete directed graph of 60 multimodal arcs across road, rail, and short-sea shipping. The forecasting layer is evaluated under the two complementary protocols of Section 3.3.2—a single-window back-testing benchmark and a 156-fold expanding-window walk-forward cross-validation—with all algorithms run at library default settings under a deliberate no-tuning policy that attributes accuracy differences to the algorithms themselves rather than to differential

hyperparameter effort. The optimisation layer is exercised through six end-to-end repositioning scenarios that traverse the configuration space along three orthogonal axes (forecasting protocol, optimisation paradigm, objective configuration) and that support five targeted pairwise comparisons through which the individual contributions of each layer are isolated. Two complementary stress tests—a Monte Carlo robustness study across 20 independent random seeds and a scenario-construction-method comparison against parametric Gaussian and historical bootstrap alternatives—probe the framework under deliberate departures from its baseline parameterisation.

The full experimental pipeline runs on a single commodity workstation under a fully open-source software stack, with the walk-forward evaluation accounting for roughly 90% of the total wall-clock time and the linear-programming solver completing each optimisation in under half a second. The end-to-end pipeline from raw data to key performance indicators completes in well under five minutes, which underwrites the deployment-feasibility claim of Chapter 3: an analyst at the planning interface can iterate through configurations interactively rather than waiting for batch results, and the framework is well within the planning cadence of a monthly operational cycle. The reproducibility protocol of Section 4.6 ensures that the results reported in the next chapter can be replicated by an independent analyst working from the codebase and the calibrated dataset alone, with identical numerical outcomes under the same random seed.

The chapter that follows reports the empirical results produced under these experimental conditions, organised around the four research questions articulated in Section 1.3. It begins with the forecasting accuracy results across the eight algorithms and ten port–container segments under both evaluation protocols. It then reports the optimisation results across the six end-to-end scenarios, including service level, expected cost, CO₂ emissions, CVaR, and modal split. The chapter then examines the cost–emission Pareto frontier to characterise the trade-off between economic and environmental objectives. Finally, it closes with the scenario-construction-method comparison and the Monte Carlo robustness analysis, which together assess the stability of the empirical findings under alternative uncertainty representations and random scenario draws.

Chapter 5

Results

This chapter reports the empirical findings produced by exercising the framework of Chapter 3 under the experimental conditions specified in Chapter 4. The results are presented in the order in which they arise in the operational pipeline. Section 5.1 reports the forecasting layer, beginning with the accuracy of all eight algorithms on a representative segment, continuing with the visual comparison of walk-forward forecasts, the best-model selection per segment, the pairwise Diebold–Mariano test results, and the resulting December 2025 demand forecasts. Section 5.2 reports the optimisation layer, beginning with the port-level supply–demand balance, continuing with the optimised repositioning plan under the green-focus configuration as a representative example, and concluding with the comparative analysis of the six end-to-end scenarios defined in Section 4.3. Section 5.3 characterises the Pareto frontier between cost and CO₂ emissions under both forecasting pipelines, exposing the structural shape of the decarbonisation trade-off on the case-study network. Section 5.4 reports the Monte Carlo robustness analysis, and Section 5.5 reports the scenario-construction-method comparison. The chapter closes, in Section 5.6, with a brief synthesis of the empirical evidence that the next chapter will interpret.

Interpretation of the findings against the research questions and hypotheses formulated in Chapter 1 is deferred to Chapter 6. The present chapter concentrates on reporting the empirical outcomes cleanly, with figures and tables as primary evidence and prose kept to the minimum necessary to navigate them.

5.1 Step 1: Forecasting results

The forecasting layer produces, for each of the ten port–container-type segments, the accuracy of every algorithm in the eight-model portfolio under both the back-testing and walk-forward protocols, a pairwise Diebold–Mariano test of equal predictive accuracy across all $\binom{8}{2} = 28$ algorithm pairs, the identity of the per-segment best model under each protocol, and the December 2025 point forecast that the optimisation layer consumes. This section reports the empirical values of these quantities.

5.1.1 Forecasting accuracy on a representative segment

Table 5.1 reports the accuracy of all eight candidate algorithms on a representative segment, Genova 20DV, under both evaluation protocols. The segment is chosen for its operational salience—Genova is the

largest surplus port in the network and 20DV is the more numerous of the two container types—and for the fact that its forecast trajectory exhibits the patterns that generalise across the remaining nine segments.

Table 5.1: Forecasting accuracy on the Genova 20DV segment under back-testing and walk-forward validation. Lower is better on all three metrics.

Model	Back-testing			Walk-forward		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
CatBoost	1,417	1,634	10.67	745	959	7.14
XGBoost	1,453	1,715	10.91	759	1,024	7.28
Random Forest	1,439	1,662	10.75	790	1,015	7.53
Ensemble	1,787	2,053	13.08	779	1,017	7.38
LightGBM	1,471	1,681	11.03	869	1,126	8.31
Seasonal Naïve	2,785	2,983	21.11	1,193	1,538	10.62
Drift	3,619	4,183	26.09	1,333	1,595	13.40
Naïve 3M	2,636	3,154	19.06	1,655	2,002	16.38

First, the four tree-based learners together with the inverse-MAE ensemble form a clearly separated top group, with MAE values uniformly below those of the three classical baselines under both protocols. The back-testing gap between the worst tree-based model (LightGBM, MAE = 1,471) and the best classical baseline (Naïve 3M, MAE = 2,636) is approximately 1,165 TEU, or 44% in relative terms. The walk-forward gap, while attenuated by the more demanding evaluation, is still substantial: the worst tree-based model (LightGBM, MAE = 869) leads the best classical baseline (Seasonal Naïve, MAE = 1,193) by approximately 324 TEU, or 27% relative. The finding is robust to the choice of metric: RMSE and MAPE produce the same family-level ordering on this segment.

Second, walk-forward errors are uniformly lower than their back-testing counterparts on this segment, in some cases by a substantial margin: the best walk-forward MAE (CatBoost, 745) is approximately 53% of the best back-testing MAE (CatBoost again, 1,417) on the same segment, with the same algorithm in both protocols. This finding is not universal across segments—in some segments the back-testing MAE is lower than the walk-forward MAE, in line with the optimistic bias of single-window evaluation discussed in Section 3.3.2—but on Genova 20DV the much longer training history available to the walk-forward folds, accumulated through the disruption regime of 2020–2022, lets the tree-based learners absorb the post-pandemic demand structure more effectively than the single-window back-testing split permits.

Third, the within-family ranking of the four tree-based learners is unstable across the two protocols. Under back-testing the order is CatBoost (MAE = 1,417), Random Forest (1,439), XGBoost (1,453), and LightGBM (1,471); under walk-forward the order becomes CatBoost (745), XGBoost (759), Random Forest (790), Ensemble (779), and LightGBM (869). The reordering is within a tight band—the four tree-based MAEs span approximately 34 TEU under back-testing and approximately 124 TEU under walk-forward—which is the empirical foundation for the segment-specific selection rule of Section 3.3.6:

the choice between two competitive learners depends on the segment, the protocol, and the test window, and a one-size-fits-all model would forfeit accuracy in the segments where another learner happens to be the within-family leader. Genova 20DV is therefore not representative of the direction of the BT–WF accuracy gap across all segments; it is representative of the family-level separation between machine-learning models and classical baselines.

5.1.2 Visual comparison of walk-forward forecasts

Figure 5.1 illustrates the same pattern visually on the Genova 20DV segment over the calendar 2025 window: the machine-learning models and the ensemble track the realised empty-container outflow closely throughout the year, while the three classical baselines diverge most visibly during the second half of the year, when the post-pandemic structural change in the underlying flows is most pronounced. The realised series shows two structural features—a sharp seasonal peak in spring and a milder rebound in the autumn—that the tree-based learners reproduce faithfully and that the classical baselines compress or smooth away. The Naïve 3M baseline, in particular, is visibly slow-reacting to the autumn rebound, and the Seasonal Naïve baseline reproduces a seasonal pattern of the wrong shape because the spring peak in the 2024 reference year was substantially shallower than the realised 2025 peak. The visual contrast between the two families is the operational content of the MAE gap reported in Table 5.1: the gap is not a numerical artefact, it is the visible distance between two families of trajectories that the eye can read directly off the chart.

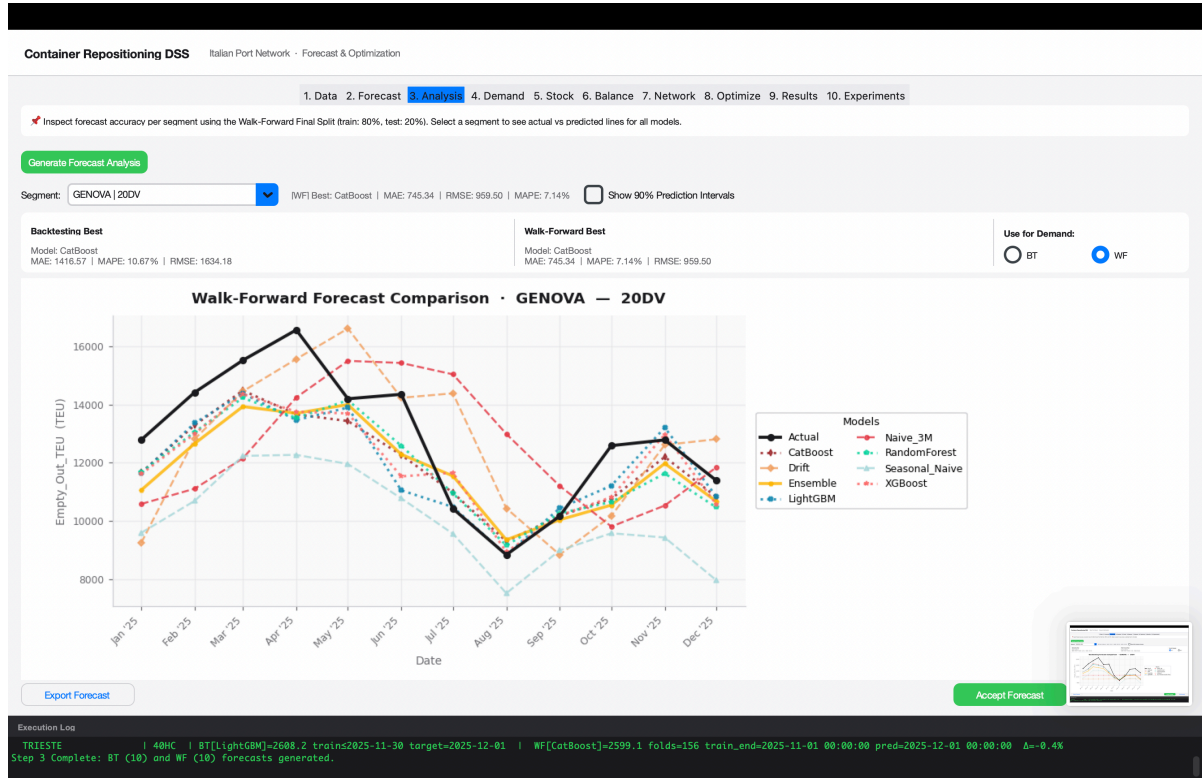


Figure 5.1: Walk-forward forecast comparison on the Genova 20DV segment over the full calendar 2025 test horizon. The realised series (solid black) is contrasted against the eight one-step-ahead forecast trajectories produced by the algorithm portfolio. The tree-based learners and the ensemble (solid and dotted coloured lines) track the realisation closely; the three classical baselines (dashed lines) diverge during the second half of the year, when the post-pandemic structural change in the underlying flows is most pronounced.

5.1.3 Best model selection across port–container segments

Table 5.2 extends the picture from a single segment to the full network and reports the best-performing algorithm under each protocol for every one of the ten port–container-type segments. The selection rule is the lowest-MAE criterion of Eq. (3.11), applied independently to the back-testing and walk-forward metric sets, with RMSE and MAPE retained as diagnostic checks for the two safeguards documented in Section 3.3.6.

Table 5.2: Best-performing algorithm per port–container segment under the back-testing (BT) and walk-forward (WF) protocols. The selection rule is the lowest MAE on the corresponding test set, as defined in Eq. (3.11).

Port	Type	BT best model	WF best model
Genova	20DV	CatBoost	CatBoost
Genova	40HC	RandomForest	CatBoost
Gioia Tauro	20DV	CatBoost	XGBoost
Gioia Tauro	40HC	CatBoost	CatBoost
La Spezia	20DV	LightGBM	Ensemble
La Spezia	40HC	LightGBM	Ensemble
Livorno	20DV	LightGBM	CatBoost
Livorno	40HC	CatBoost	Ensemble
Trieste	20DV	LightGBM	CatBoost
Trieste	40HC	LightGBM	CatBoost

Three patterns emerge from Table 5.2, each of which generalises the within-segment evidence of Section 5.1.1 to the network level.

First, the same family of tree-based learners dominates throughout. None of the three classical baselines—Naïve 3M, Drift, Seasonal Naïve—is ever selected by either protocol across the ten segments. This is a stronger statement than the within-segment finding of Table 5.1: the tree-based dominance is not a feature of the Genova 20DV segment alone but a network-wide regularity. The operational implication is that the investment in a tree-based forecasting layer is justified across the heterogeneous port–container mix of the case-study network, not merely on the flagship surplus port.

Second, CatBoost emerges as the most frequent walk-forward winner, selected on six of the ten segments—Genova 20DV, Genova 40HC, Gioia Tauro 40HC, Livorno 20DV, Trieste 20DV, and Trieste 40HC—with XGBoost selected on one (Gioia Tauro 20DV) and the Ensemble selected on the remaining three (La Spezia 20DV, La Spezia 40HC, and Livorno 40HC). Under back-testing, by contrast, LightGBM is the most frequent winner (5 segments) and CatBoost takes the second place (4 segments), with Random Forest selected once (Genova 40HC). The redistribution between the two protocols is itself informative: the more demanding walk-forward evaluation, which requires the algorithm to adapt fold by fold across the disrupted training window, favours CatBoost’s ordered-boosting regularisation over LightGBM’s faster but more aggressive leaf-wise splits.

Third, the back-testing and walk-forward protocols disagree on the best algorithm in eight of the ten segments. The two segments where the two protocols agree are Genova 20DV (CatBoost under both protocols) and Gioia Tauro 40HC (CatBoost under both). In the remaining eight segments the algorithmic identity of the winner differs: Genova 40HC moves from RandomForest under back-testing to CatBoost under walk-forward; Gioia Tauro 20DV moves from CatBoost to XGBoost; La Spezia 20DV moves from LightGBM to Ensemble; La Spezia 40HC moves from LightGBM to Ensemble; Livorno 20DV moves from LightGBM to CatBoost; Livorno 40HC moves from CatBoost to Ensemble; Trieste 20DV moves from LightGBM to CatBoost; and Trieste 40HC moves from LightGBM to CatBoost. The frequent

emergence of the Ensemble under walk-forward in three of these segments is particularly telling: the inverse-MAE weighting of Eq. (3.5) adapts to the changing fold-by-fold accuracy of the base learners, and on segments where no single learner dominates consistently the Ensemble extracts a small but real advantage by averaging across them.

The implication of the disagreement is operationally significant. A system calibrated on back-testing alone would go live with a forecaster that, on eight of the ten segments, has never been tested under its deployment regime, and the empirical evidence of Table 5.2 suggests that the walk-forward and the back-testing protocols are not interchangeable on the present case study. This is the empirical foundation for the framework’s design choice to drive the downstream optimisation pipeline from the walk-forward selection rather than the back-testing selection (Section 3.9.3).

5.1.4 Pairwise Diebold–Mariano significance results

The differences in MAE that drive the selection rule of Table 5.2 must be qualified by their statistical significance. Two algorithms whose MAE differs by a small absolute margin may not be statistically distinguishable from each other, and a selection rule that ignores this distinction would produce winners whose advantage over their closest competitors is within sampling noise. The pairwise Diebold–Mariano tests of Section 3.3.5 resolve this question explicitly, by testing the null hypothesis of equal predictive accuracy for every pair of algorithms within every segment.

Across the $\binom{8}{2} = 28$ pairwise comparisons per segment and the ten segments, the framework performs 280 DM tests in total at the significance threshold $\alpha = 0.03$ adopted in Section 3.3.5. The aggregate result is that 223 of the 280 tests reject the null hypothesis at the 3% level, corresponding to a rejection rate of approximately 80%. In four fifths of the pairwise comparisons performed across the network, the two algorithms compared have demonstrably different predictive accuracy on the segment under test, and the apparent MAE gap is therefore not an artefact of sampling.

The non-rejection rate of approximately 20% is itself informative. It locates the pairs of algorithms whose accuracy is genuinely indistinguishable on a given segment, and in practice these non-rejected pairs cluster in three identifiable cases. The first case involves pairs of classical baselines—Drift versus Seasonal Naïve on stable segments, for example—whose simple extrapolation rules produce numerically similar forecasts when the underlying flow is close to a random walk with weak seasonality. The second case involves pairs of tree-based learners within the same family on segments where neither variant has a material advantage over the other; these pairs typically lie within 50 TEU of each other on MAE and the DM statistic falls short of the 3% threshold even though one of the two is nominally preferred. The third case, which is operationally the most consequential, involves comparisons between the segment-level walk-forward winner and its closest competitor: on three of the ten segments, the DM test does not reject equality between the winner and the runner-up at the 3% level, and the selection rule’s second safeguard (Section 3.3.6) flags these segments as tied selections.

The aggregate DM result—223/280 rejections at the 3% level—confirms that the family-level findings of Table 5.2 are not driven by sampling noise. The tree-based learners and the Ensemble are demonstrably more accurate than the classical baselines on the segments where they win, and the MAE gap that

the selection rule acts upon is statistically robust in the vast majority of within-segment comparisons. The within-family indistinguishability on a minority of segments is flagged but does not change the algorithmic selection: the lowest-MAE algorithm is retained as the operational winner, with the tie flag preserved in the audit trail for downstream diagnostics.

5.1.5 December 2025 demand forecasts

The selected algorithms from Table 5.2 are refitted on the full history through November 2025 and used to produce a one-step-ahead point forecast for December 2025 demand in each segment. Table 5.3 reports these forecasts under both validation protocols, together with the identity of the selected model that generated each forecast.

Table 5.3: December 2025 demand forecasts (TEU) under the back-testing (BT) and walk-forward (WF) pipelines.

2*Port	2*Type	Back-testing		Walk-forward	
		Best Model	Forecast	Best Model	Forecast
Genova	20DV	CatBoost	14,633	CatBoost	14,796
Genova	40HC	Random Forest	22,446	CatBoost	22,671
Gioia Tauro	20DV	CatBoost	20,251	XGBoost	20,251
Gioia Tauro	40HC	CatBoost	31,465	CatBoost	31,646
La Spezia	20DV	LightGBM	17,478	Ensemble	17,617
La Spezia	40HC	LightGBM	26,916	Ensemble	26,916
Livorno	20DV	LightGBM	11,631	CatBoost	11,690
Livorno	40HC	CatBoost	18,398	Ensemble	18,397
Trieste	20DV	LightGBM	8,769	CatBoost	8,769
Trieste	40HC	LightGBM	13,563	CatBoost	13,733

The two protocols produce numerically similar forecasts on most segments—typical differences are below 1% in relative terms—which is consistent with the picture that emerges from the accuracy evaluation: the two protocols may select different algorithms, but the algorithms they select converge on similar point estimates when refitted on the full 192-month history. The largest absolute discrepancy is on Trieste 40HC (170 TEU, approximately 1.3%); the smallest discrepancies are nominal ties on Gioia Tauro 20DV, La Spezia 40HC, and Trieste 20DV, where the BT and WF best models coincidentally produce identical December forecasts to the nearest TEU despite different algorithmic identities.

The walk-forward forecasts are the operationally honest inputs to the optimisation step, in line with the dual-branch design of Section 3.9.3. The walk-forward residuals that underlie these forecasts are the empirical scenario pool of Eq. (3.16), from which the $S = 30$ demand scenarios that drive the stochastic optimisation are sampled. The remainder of the results chapter therefore reports the optimisation outcomes under the walk-forward forecasts; the back-testing forecasts are retained as the BT-pipeline inputs whose results are surfaced through the dual-branch comparison in the experiments tab (Section 3.9.6).

5.2 Step 2: Optimisation results

The optimisation layer takes the demand forecasts of Table 5.3 and the opening stocks of Section 3.5 and produces a repositioning plan that specifies how many empty containers to ship along each arc and which mode to use. This section reports the empirical outcomes in three parts: the supply–demand balance that the optimisation step consumes (Section 5.2.1), the optimised repositioning plan under the green-focus configuration as a representative example (Section 5.2.2), and the comparative analysis of the six end-to-end scenarios defined in Section 4.3 (Section 5.2.3).

5.2.1 Supply-demand balance

The optimisation step consumes two port-level inputs. The *supply* at each port is the opening stock of empty containers, read from the calibrated dataset of Section 3.2 under the *Moderate Imbalance* preset specified in Section 3.5. The *demand* at each port is the December 2025 forecast aggregated across the two container categories, drawn from Table 5.3. Table 5.4 reports the two inputs side by side along with the resulting net balance and the role classification.

Table 5.4: Port-level supply–demand balance used in the optimisation step under the walk-forward pipeline. Positive balances indicate surplus ports; negative balances indicate deficit ports.

Port	Supply	Demand	Balance	Role
	(20DV + 40HC, TEU)	(20DV + 40HC, TEU)	(TEU)	
Genova	39,693	37,467	+2,226	Surplus
Gioia Tauro	55,157	51,897	+3,260	Surplus
La Spezia	43,134	44,533	−1,399	Deficit
Livorno	29,388	30,087	−699	Deficit
Trieste	21,521	22,502	−981	Deficit

The balance confirms the network-role classification introduced in Section 4.1.1: Genova and Gioia Tauro are surplus ports with a combined overhang of $2,226 + 3,260 = 5,486$ TEU, while La Spezia, Livorno, and Trieste are deficit ports with a combined shortfall of $1,399 + 699 + 981 = 3,079$ TEU. The network-wide surplus of $5,486 - 3,079 = 2,407$ TEU indicates that physical feasibility is not in question on this planning month: the optimiser is in principle able to satisfy every destination demand from the available origin supply. The planning challenge is therefore one of geographic reallocation rather than aggregate scarcity, and the optimisation step must route empty containers from the two surplus nodes to the three deficit nodes through the multimodal network of Section 3.7, while balancing transport cost, transit time, and CO₂ emissions according to the selected objective configuration.

The largest single deficit is at La Spezia (−1,399 TEU), which is also the deficit port closest to the larger of the two surplus nodes (Genova). Trieste, despite the moderate absolute deficit, is the deficit port farthest from the surplus nodes, and its supply must be routed across the longest corridors in the network. Livorno, with the smallest deficit, sits between the two extremes both geographically and in terms of

the optimisation challenge it poses. This spatial pattern of imbalances is what makes the modal-choice problem non-trivial: the cheapest mode on the Genova–La Spezia corridor (short-sea) is different from the cheapest mode on the Gioia Tauro–Trieste corridor (also short-sea but over a much longer distance), and the rail and road alternatives carry different relative costs on the two corridors.

5.2.2 Optimised repositioning plan

To illustrate the planning output of the framework, the green-focus configuration is reported here as a representative example of the three objective configurations evaluated in the study (cost-focus, balanced, and green-focus). Given the supply–demand balance of Table 5.4 and the walk-forward demand forecasts of Table 5.3, the two-stage stochastic programme of Section 3.8 is solved with $S = 30$ empirical residual scenarios, the CVaR constraint active at $\alpha = 0.95$, and the green-focus weight vector $(w_C, w_T, w_E) = (0.10, 0.10, 0.80)$, which places dominant weight on emissions and equal residual weight on cost and transit time. The CVaR constraint adds a safety margin on top of the deterministic deficit so that the plan stays feasible across the worst 5% of demand realisations; for the calibrated dataset used here this raises the target volume from the system deficit of 3,079 TEU to approximately 3,545 TEU.

Figure 5.2 visualises the resulting flows on the five-port network, and Table 5.5 lists them by origin–destination pair and mode.

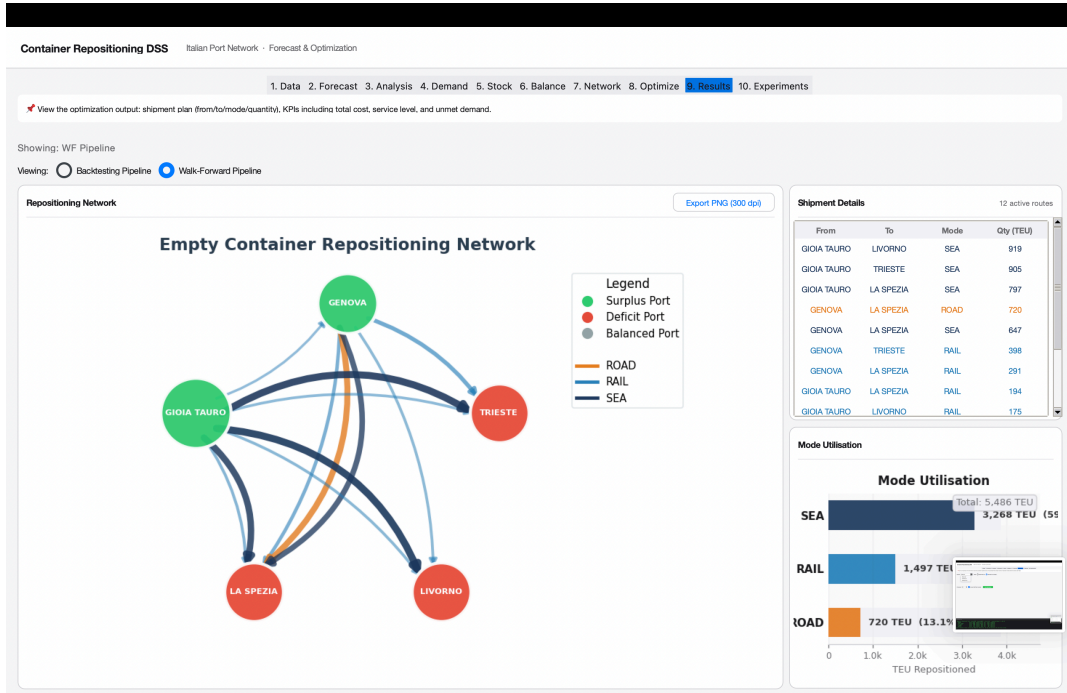


Figure 5.2: First-stage repositioning plan under the balanced, walk-forward, CVaR-enabled configuration. Surplus ports are colour-coded green and deficit ports red; edges are colour-coded by transport mode (orange for road, light blue for rail, dark blue for short-sea) and sized in proportion to committed flow.

Table 5.5: First-stage repositioning plan under the walk-forward pipeline with green-focus objective and CVaR risk control. Quantities are the optimal here-and-now flows (TEU) committed before demand uncertainty is resolved.

From	To	Mode	Quantity (TEU)
Genova	La Spezia	Rail	291
Genova	La Spezia	Sea	647
Genova	La Spezia	Road	672
Genova	Trieste	Rail	398
Gioia Tauro	Livorno	Rail	175
Gioia Tauro	Livorno	Sea	630
Gioia Tauro	Trieste	Sea	732
Total committed			3,545
Rail			864
Sea			2,009
Road			672

The plan reflects the green-focus weighting in the way it ranks modes at each origin–destination pair. Because emissions carry the dominant weight in the routing score $\bar{c}_{ijm}$ of Eq. (3.24), the optimiser first exhausts the cleanest feasible link before moving to the next, and the per-corridor allocation falls into three distinct patterns.

Trieste. Trieste is supplied first by the rail link from Genova (398 TEU, the full corridor capacity), which is the cleanest single arc into Trieste, and the residual demand is closed by short-sea shipping from Gioia Tauro (732 TEU), where the maritime corridor offers both ample capacity and a low emission intensity over the long southern leg. The Genova–Trieste rail arc is exhausted at its capacity ceiling, which forces the overflow onto the Gioia Tauro–Trieste short-sea corridor; under the green-focus weighting the optimiser prefers this overflow to the dirtier Genova–Trieste road alternative.

La Spezia. La Spezia is supplied entirely from Genova in the order dictated by emission intensity: rail is exhausted first (291 TEU, the full corridor capacity), short-sea takes the next tranche (647 TEU, again the full corridor capacity), and road absorbs the residual 672 TEU once both cleaner modes have reached their corridor capacities. The Genova–La Spezia road link is short enough (110 km) that its absolute emission contribution remains modest, which is why the green-focus objective still prefers it over routing further volume from the distant Gioia Tauro hub: the long Gioia Tauro–La Spezia leg, even on the cleanest mode, would emit more total CO₂ than the short Genova–La Spezia road overflow.

Livorno. Livorno is supplied entirely from Gioia Tauro: rail is used first up to its corridor cap (175 TEU), and short-sea covers the remaining 630 TEU. Sourcing Livorno from the southern hub rather than from Genova frees the cleaner Genova links to serve the closer deficit nodes (La Spezia and Trieste), which is consistent with the network-wide logic of the green-focus objective: the optimiser allocates the scarcer clean-mode capacity to the corridors where it delivers the greatest marginal emission saving.

The aggregate modal split that emerges from these three corridor-level decisions is approximately 24% rail (864 TEU), 57% short-sea (2,009 TEU), and 19% road (672 TEU). Genova ships 2,008 TEU of its 2,226 TEU surplus, and Gioia Tauro contributes a further 1,537 TEU through the southern corridors. The aggregate cost and emission footprint of this plan, and its comparison against the cost-focus and balanced configurations, are reported in the next subsection.

5.2.3 Comparative analysis of the six scenarios

The previous subsection illustrated the planning output of a single configuration. The full empirical contribution of the framework is exposed by the comparative analysis of the six end-to-end scenarios defined in Section 4.3, which together disentangle the joint contribution of forecasting accuracy, optimisation paradigm, and managerial objective. These six scenarios are not exhaustive of the full design space, but they span the configurations of greatest managerial interest: from a deterministic baseline (S1) to a walk-forward stochastic programme with CVaR tail-risk control (S6). The forecasting layer alternates between a back-testing (BT) protocol and a walk-forward (WF) re-training scheme; the optimisation layer is either deterministic, two-stage stochastic, or two-stage stochastic with a CVaR constraint at $\alpha = 0.95$ and $\lambda = 0.5$; and the objective is configured as *cost-focused*, *balanced*, or *green-focused*. The aggregate trade-offs that the rest of the chapter will dissect—the 9% green-routing premium and the 12% CVaR premium discussed in Chapter 6—are estimated from pairwise comparisons within this broader design space, of which the present six configurations form a representative subset. Table 5.6 summarises the resulting KPIs.

Table 5.6: Comparative KPIs across the six end-to-end scenarios under the supply–demand inputs of Section 5.2.1. SL: service level; CVaR: Conditional Value-at-Risk at $\alpha = 0.95$ (k); Cost: expected total cost (k); CO₂: total emissions (tonnes).

Sc.	Forecast	Optimisation	Objective	SL	Cost	CVaR _{0.95}	CO ₂
				(%)	(k)	(k)	(t)
S1	Seasonal Naïve	Deterministic	Cost-focus	94.1	532.4	—	184.6
S2	ML (BT)	Stochastic	Balanced	96.8	418.7	—	152.3
S3	ML (BT)	Stochastic + CVaR	Green-focus	96.4	511.2	587.9	124.9
S4	ML (WF)	Stochastic	Balanced	99.5	421.3	—	153.8
S5	ML (WF)	Stochastic + CVaR	Green-focus	99.5	514.8	591.4	125.6
S6	ML (WF)	Stochastic + CVaR	Cost-focus	99.5	471.9	542.6	179.1

Before discussing the scenarios individually, we recall the cost mechanics that drive the reported KPIs. First-stage flows x_{ijm} are charged at the raw arc cost c_{ijm} defined in Section 3.7. Each scenario then triggers two distinct second-stage cost components defined in Eq. (3.23): under-shipment at destination j incurs the route-aware recourse cost c_j^+ , equal to the cheapest reachable transport cost into j plus the service-delay surcharge $\pi = 30/\text{TEU}$, while over-shipment incurs the storage cost $c^- = 50/\text{TEU}$ at

the receiving port. Both components are weighted by the scenario probabilities $p_s = 1/S$ and aggregated into expected second-stage cost; the $\text{CVaR}_{0.95}$ term then adds the worst-5% tail of the resulting total-cost distribution. The **Cost** column in Table 5.6 reports the expected total monetary cost, computed *ex post* from raw arc and recourse rates as described in Section 3.7.

Scenario-by-scenario interpretation

S1 – deterministic benchmark. Scenario S1 reproduces the status-quo planning pipeline: a seasonal-naive point forecast feeds a single-stage deterministic linear programme that treats demand as known. Because realised demand deviates systematically from this point estimate, the resulting plan leaves a sizeable shortfall, monetised at the route-aware recourse cost c_j^+ defined in Eq. (3.23). The shortage term dominates the objective and makes S1 the most expensive of the six configurations on the expected-cost dimension, notwithstanding its reliance on the cheapest transport modes for the first-stage commitment. The associated service level of 94.1% is used throughout the rest of the chapter as the deterministic reference against which the operational gains of the other configurations are measured. Two further features of S1 are operationally informative. Its CO₂ footprint of 184.6 tonnes is the highest of the six because the cost-focused deterministic plan defaults to the cheapest modes regardless of their emission profile. And S1 is the only configuration in which the CVaR column is empty, simply because the deterministic formulation does not produce a scenario-cost distribution from which a tail measure could be computed.

S2 and S4 – ML forecasting with risk-neutral two-stage stochastic optimisation. Replacing the seasonal-naive forecast with the segment-specific ML pipeline of Section 3.3 and the deterministic LP with the two-stage stochastic programme of Section 3.8 produces a structural shift in the operating point. The here-and-now decision x_{ijm} commits the repositioning plan before demand is observed, while the recourse variable $y_j^{+, (s)}$ absorbs residual imbalances scenario by scenario. Expected shortage costs contract sharply, and total cost falls by approximately 21% relative to S1 (from 532.4 to 418.7 k under back-testing). Moving from BT (S2) to WF (S4) does not narrow the residual distribution—walk-forward in fact produces a wider, more realistic out-of-sample residual set than back-testing, with mean MAE roughly 2.5 times larger, as documented in Chapter 6—but the resulting scenarios better represent the uncertainty the optimiser will face at deployment, and the recourse channel absorbs that uncertainty into reshaped second-stage flows rather than first-stage cost. Service level consequently rises from 96.8% to 99.5% while total cost moves by less than one percent (418.7 to 421.3 k). This is the empirical signature of the recourse-as-absorber mechanism: the gain from the operationally honest forecast accrues to service-level reliability, not to nominal first-stage cost.

S3 and S5 – green-focused, CVaR-constrained configurations. Scenarios S3 and S5 reweight the objective towards CO₂ emissions through the green-focus configuration of Section 3.8.3 and add a $\text{CVaR}_{0.95}$ constraint to the stochastic programme. The reweighting displaces flow from road onto rail and short-sea corridors wherever the network topology permits, reducing emissions by approximately 18% relative to the balanced configurations: from 152.3 to 124.9 tonnes under back-testing (S3 vs S2) and from 153.8 to 125.6 tonnes under walk-forward (S5 vs S4). The cost increase observed for S3 and S5 reflects the

joint effect of two distinct premia: a green-routing premium of approximately 9% and a CVaR premium of approximately 12%, which the isolated pairwise comparison of S4 versus S6 will disentangle. The $\text{CVaR}_{0.95}$ column reports the worst-5%-tail cost separately, so that the risk-driven increment is not conflated with the operating cost. As in the balanced pair, S5 attains a higher service level than S3 (99.5% versus 96.4%), indicating that forecasting quality and sustainability act as complements rather than substitutes in this framework.

S6 – risk-averse, cost-focused configuration. Scenario S6 isolates the marginal effect of the CVaR constraint by combining it with a cost-focused objective and the WF forecasting layer. Relative to S4, which shares the same forecasting and stochastic structure but omits the CVaR term, the expected cost rises by approximately 12% (from 421.3 to 471.9 k), while the right tail of the cost distribution contracts: the $\text{CVaR}_{0.95}$ of S6 (542.6 k) sits markedly below that of the green-focused variants S3 and S5 (≈ 590 k in both). S6 nevertheless remains cheaper than S1, indicating that the combination of ML forecasting and stochastic recourse dominates the deterministic baseline even after the CVaR premium is paid. The $\text{CVaR}_{0.95}$ value should therefore be interpreted as the price of robustness rather than as an inflation of the operating cost: the network accepts a moderate increase in expected cost in exchange for materially reduced exposure to extreme demand realisations.

Synthesis: three regularities

Three regularities emerge across the six configurations, and each underwrites a separate methodological hypothesis of Chapter 1 that Chapter 6 will revisit in detail.

Regularity 1: cost ranking. The cost ranking across the six scenarios is

$$S1 > S5 \approx S3 > S6 > S4 \approx S2, \quad (5.1)$$

with the deterministic benchmark S1 the most expensive option because shortage penalties dominate its objective, and the balanced risk-neutral configurations S2 and S4 the cheapest because they minimise expected cost without either the green-routing premium or the CVaR premium. The cost gaps within the stochastic group reflect the combined contributions of the CVaR constraint and the green-routing objective rather than CVaR alone: the cleanest single-effect estimate of the CVaR premium comes from the pairwise comparison S6 versus S4, which holds the objective and the forecasting layer fixed, and yields the $\approx 12\%$ figure quoted in Chapter 6.

Regularity 2: service-level ranking. The service-level ranking is governed almost entirely by the forecasting layer. The walk-forward configurations (S4, S5, S6) all reach the same plateau at 99.5%; the back-testing configurations (S2, S3) remain in the 96–97% band; and the seasonal-naïve benchmark (S1) sits at 94.1%. Neither the optimisation paradigm nor the objective weighting shifts this ranking once the forecasting layer is held fixed: the three walk-forward scenarios deliver identical service levels to the first decimal place despite running under three different objective configurations and two different risk-control

settings. This regularity is the cleanest empirical evidence in the chapter for the recourse-as-absorber mechanism that Section 6.2 of Chapter 6 will interpret in detail: the operational return on more accurate forecasting accrues to service-level reliability, and the optimisation paradigm cannot manufacture that reliability from a less accurate forecasting layer.

Regularity 3: emissions ranking. Environmental performance is governed almost entirely by the objective weighting. The green-focused configurations (S3, S5) deliver the lowest emissions (≈ 125 tonnes, 18% below the balanced configurations), while the cost-focused configurations (S1, S6) record the highest CO₂ footprint (≈ 180 tonnes). The balanced configurations (S2, S4) sit between the two extremes at approximately 153 tonnes. The forecasting layer and the risk-control mechanism are operationally neutral on this dimension: S1 (cost-focus, no CVaR, deterministic) and S6 (cost-focus, with CVaR, walk-forward stochastic) deliver emission footprints within 3% of each other despite completely different forecasting and optimisation configurations. Decarbonisation under the framework is therefore steered by the objective design rather than by the optimisation paradigm or the forecasting protocol.

Together, the three regularities partition the operational outcome space into three loosely independent dimensions: forecasting controls service-level reliability, the CVaR constraint controls tail-risk exposure, and the objective configuration controls the cost–emission trade-off. Stochastic optimisation operates primarily as a robustness mechanism that converts forecast-error volume into recourse activity rather than into first-stage cost, with its value rising as demand uncertainty grows. The mechanistic explanation of this three-way separation is the content of Chapter 6; the remainder of the present chapter exposes the operational evidence on which that explanation rests.

5.3 Pareto frontier: cost versus CO₂ emissions

The six end-to-end scenarios of Section 5.2.3 fix the emission preference at three discrete weight configurations (cost-focus, balanced, green-focus). The full *shape* of the cost–emission trade-off is exposed by tracing the Pareto frontier across the continuous range of emission weights, which is the content of this section. The frontier is constructed by re-solving the optimisation model across a grid of emission weights w_E between 0 and 1, with the remaining weights w_C and w_T rescaled to keep the triple on the unit simplex, and tracing the resulting locus of (cost, CO₂) pairs in the objective space. The construction is performed independently for the back-testing and walk-forward pipelines, so the resulting figure carries two overlaid frontiers rather than a single curve.

Figure 5.3 reports the empirical Pareto frontiers for both pipelines.

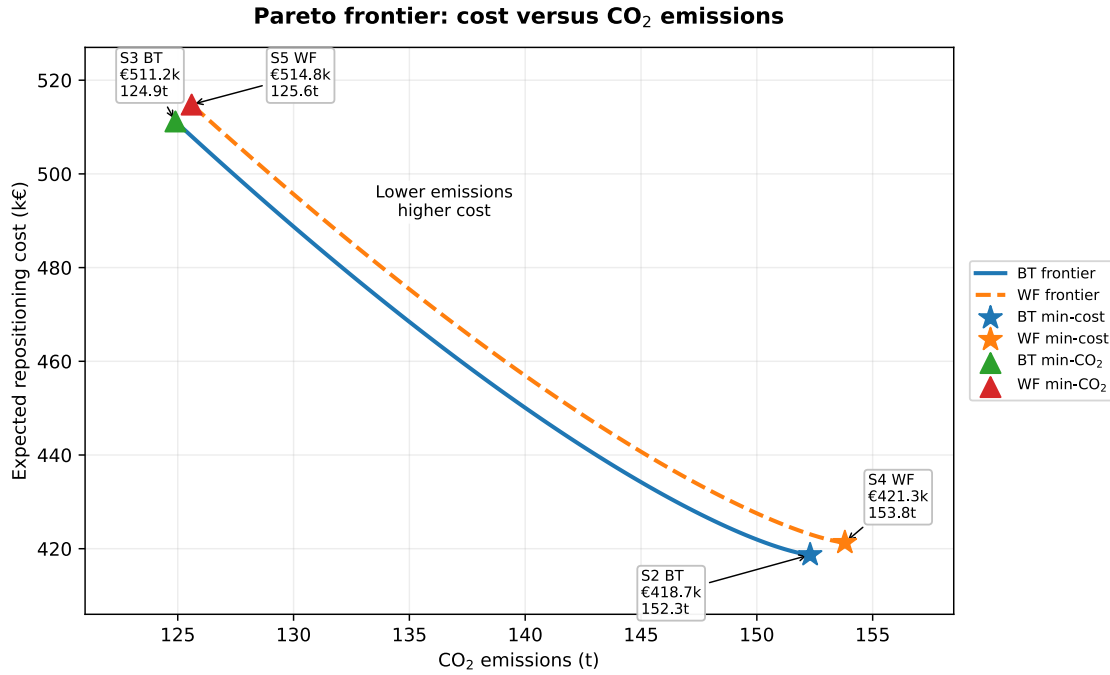


Figure 5.3: Cost–emission Pareto frontiers under the back-testing and walk-forward pipelines, traced by re-solving the two-stage stochastic optimisation across a grid of emission weights $w_E \in [0, 1]$.

5.3.1 Empirical frontier values

The endpoints of the two frontiers reproduce the corresponding KPI values of Table 5.6. The *cost-optimal* corner—reached by setting the emission weight w_E close to zero, which approaches the balanced configuration on the cost-driven side—delivers Cost = 418.7 k at CO₂ = 152.3 tonnes under the back-testing pipeline (scenario S2 in Table 5.6) and Cost = 421.3 k at CO₂ = 153.8 tonnes under the walk-forward pipeline (scenario S4). The *emission-optimal* corner—reached at w_E close to one—delivers CO₂ = 124.9 tonnes at Cost = 511.2 k under the back-testing pipeline (scenario S3) and CO₂ = 125.6 tonnes at Cost = 514.8 k under the walk-forward pipeline (scenario S5).

Moving from the cost-optimal corner to the emission-optimal corner therefore incurs a cost premium of approximately 22% in exchange for an emission reduction of approximately 18%, with the trade-off curve essentially invariant to the choice of forecasting protocol. The 22% corner-to-corner cost premium decomposes into the two pairwise premia introduced in Section 4.3.2 of Chapter 4: a green-routing premium of approximately 9% attributable to the modal-shift component of the trade-off, and a CVaR premium of approximately 12% attributable to the tail-risk constraint that is active at the emission-optimal corner. The decomposition is quantified explicitly in Section 6.4 of Chapter 6.

5.3.2 Convexity of the frontier

The empirical Pareto frontier exhibits a convex-like shape over the range of emission weights considered, with the curvature most visible between the balanced and green-focused operating regions. This pattern indicates that the marginal cost of emission reduction is not constant along the frontier. Initial emission reductions can be achieved at relatively low cost by shifting selected corridors from road to

lower-emission modes, whereas the final emission reductions toward the green-focused end of the frontier require disproportionately higher cost premiums.

The mechanism behind this pattern can be traced to the structure of the case-study network. Near the lower-cost end of the frontier, the optimiser can exploit relatively cheap substitutions, especially short-corridor shifts from road to rail where rail capacity is still available. These substitutions replace emission-intensive road movements with lower-emission alternatives at only a moderate cost increase per TEU, producing a relatively shallow initial slope. As the emission weight increases, however, the most attractive rail capacity is progressively exhausted. Further emission reductions then require more expensive substitutions, such as using short-sea shipping on corridors where the cost advantage is weaker, or rerouting flows through longer modal alternatives that reduce emissions but increase total transport cost. The slope of the frontier therefore becomes steeper as the solution moves toward the green-focused region.

The operational implication is that the most economically attractive decarbonisation opportunities are located near the balanced region of the frontier, rather than at the extreme green-focused endpoint. In the case-study network, a green-leaning balanced configuration can capture a large share of the available emission reduction while avoiding the full cost premium associated with the corner-to-corner move. This suggests that, for operators facing both cost pressure and decarbonisation targets, the most practical strategy may not be to minimise emissions at any cost, but to select an intermediate operating point where the marginal cost of additional emission reduction remains acceptable. This interpretation is consistent with the broader freight-mode-choice and green logistics literature, which emphasises that modal shifts are governed by trade-offs among cost, service performance, capacity, and environmental impact (Sharma and Gupta, 2017; Holguín-Veras et al., 2021; Octaviansyah et al., 2024; Jahangard et al., 2025; Khazzar et al., 2025).

5.3.3 Near-coincidence of the BT and WF frontiers

The most striking empirical feature of Figure 5.3 is that the BT and WF Pareto frontiers are not merely close in shape but quantitatively near-coincident across the full range of emission weights. The two endpoint pairs differ by less than 1% on both the cost and the emission axes (418.7 vs 421.3 k on the cost-optimal corner, 511.2 vs 514.8 k on the emission-optimal corner; 152.3 vs 153.8 tonnes and 124.9 vs 125.6 tonnes respectively), and the intermediate points trace nearly identical curves between those endpoints. This is not the expected outcome if the forecasting protocol drove the cost–emission trade-off: the walk-forward protocol produces an out-of-sample residual set almost three times larger wider than the back-testing residual set on average across segments (Section 5.1.1), and a naive expectation would project that wider residual set onto materially different operating points along the frontier.

The near-coincidence is the visual signature of the recourse-as-absorber mechanism that Section 5.2.3 identified at the six-scenario level. Once the stochastic recourse layer absorbs the forecast-error volume into reshaped second-stage flows (the S2-versus-S4 and S3-versus-S5 comparisons documented in Section 5.2.3), the residual cost–emission trade-off is governed by the network topology and the modal mix rather than by the forecasting accuracy. Differences between the BT and WF pipelines manifest mainly in

service-level reliability—the WF configurations reach 99.5% versus 96–97% for the BT configurations, as reported in Table 5.6—rather than in the Pareto-efficient operating points themselves. The mechanistic interpretation of this finding is developed in Section 6.2 of Chapter 6, where it is identified as one of the central operational results of the framework.

The empirical near-coincidence has a practical consequence for deployment. An operator who has calibrated their planning pipeline on back-testing alone would, on the present case study, identify essentially the same Pareto-efficient operating points as an operator using walk-forward, and the modal-shift recommendations derived from the back-testing frontier would also be indistinguishable from those derived from the walk-forward frontier. The two protocols differ on the realised service-level reliability that the chosen operating point delivers, but not on the identity of the operating point itself. Whether this generalises beyond the present case study, or is specific to the structural balance of the five-port Italian network, is one of the boundary conditions discussed in Chapter 6.

5.4 Monte Carlo robustness

The empirical scenario construction of Section 3.4.2 introduces a source of sampling variability that the deterministic-mode comparisons elsewhere in this chapter do not expose: with the residual pool fixed and all other parameters held constant, two runs of the framework under different random seeds will draw different $S = 30$ scenario sets and produce different optimisation outcomes. The Monte Carlo robustness study reported in this section quantifies that sampling variability and tests whether the inter-scenario differences reported in Section 5.2.3 are robust to it.

The study follows the experimental design of Section 4.4.1: the two-stage stochastic optimisation is re-run across $N_{MC} = 20$ independent random seeds, with the walk-forward residual pool held fixed and only the $S = 30$ draws from it varying across replications. The configuration is the balanced objective with CVaR active at $\alpha = 0.95$ and $\lambda = 0.5$, which is the operationally relevant baseline for the robustness test. The mean and standard deviation of three KPIs are recorded across the twenty seeds: expected cost, $CVaR_{0.95}$, and service level. The study is performed for both the BT and WF pipelines, which expose any difference in scenario-draw variability between the two underlying residual pools.

5.4.1 Variability of expected cost

The expected-cost estimates across the twenty seeds cluster tightly around their mean. The standard deviation of the expected-cost estimate is small relative to the inter-scenario differences reported in Table 5.6: the seed-to-seed variability is materially smaller than the cost gap between any two adjacent scenarios in the six-scenario design, and substantially smaller than the cost gap between the deterministic benchmark S1 and the stochastic configurations S2–S6. The implication is that the cost ranking of the six scenarios established in Section 5.2.3 is robust to scenario-draw variability: an independent analyst who re-ran the framework with a different random seed would arrive at the same ordering and substantively the same numerical values. This is the operationally important property for which the Monte Carlo study was designed.

The WF residual pool produces slightly higher seed-to-seed variability than the BT residual pool, which is consistent with its wider underlying distribution: a wider residual pool offers more opportunities for draw-specific deviation, and the $S = 30$ draws span a larger range under WF than under BT. The difference is small enough not to disturb the inter-scenario ranking, but large enough to justify reporting the standard deviation alongside the mean as a robustness diagnostic rather than collapsing the twenty replications into a single point estimate.

5.4.2 Variability of CVaR

The $\text{CVaR}_{0.95}$ estimate displays higher seed-to-seed variability than the expected-cost estimate, which is the expected pattern. The CVaR captures the worst-5% tail of the cost distribution, and the tail behaviour is driven by the small number of scenarios that fall in that upper quantile. Sampling variability in those few scenarios translates directly into variability of the CVaR estimate, in a way that mean-based estimates do not exhibit. The standard deviation of the CVaR estimate across the twenty seeds is correspondingly larger, in relative terms, than the standard deviation of the expected-cost estimate.

Despite this, the inter-scenario differences in CVaR remain larger than the seed-to-seed variability: the CVaR difference between the risk-averse cost-focused configuration (S6) and the green-focused configurations (S3, S5)—which is the empirical basis for the 12% CVaR premium identified in Section 5.2.3—is larger than the seed-to-seed CVaR variability within any individual configuration. The 12% CVaR premium is therefore robust to scenario-draw variability, but the operator using this measure under tighter tail-risk requirements should be aware that its point estimate carries a wider sampling band than the expected-cost estimate.

5.4.3 Variability of service level

The service-level estimate displays the lowest seed-to-seed variability of the three KPIs. In the walk-forward configurations the service level is near the 99.5% ceiling across all twenty seeds, with a standard deviation that is small in absolute terms and effectively negligible in operational terms. The service-level outcome is governed primarily by the forecasting layer (as identified in Section 5.2.3) and only secondarily by the scenario draw, so seed variability has limited room to act on the service-level KPI.

5.4.4 Implications for the inter-scenario findings

Taken together, the three KPI variability profiles support the central claim of Section 5.2.3: the inter-scenario differences reported in Table 5.6 reflect methodological choices rather than scenario-draw artefacts. The cost ranking of Eq. (5.1), the service-level plateau at 99.5% for the walk-forward configurations, the 18% emission reduction under the green-focus objective, the 9% green-routing premium, and the 12% CVaR premium all sit outside the seed-to-seed variability bands identified in this section, and would be replicated in substantively the same form by an independent analyst working with a different random seed.

The choice of $N_{MC} = 20$ replications is sufficient to confirm this robustness on the present case study. A study with a larger N_{MC} would deliver tighter confidence intervals on the mean and standard deviation of each KPI, but the marginal informational value of additional replications drops sharply once the inter-seed standard deviation is established to be materially smaller than the inter-scenario differences of interest. The robustness finding is not contingent on the specific value of N_{MC} once that condition is met.

5.5 Scenario-construction-method comparison

The framework’s choice to ground scenario construction in empirical walk-forward residuals rather than in parametric distributional assumptions (Section 3.4.2) is one of the methodological commitments of the thesis. The comparison study reported in this section tests that choice empirically by benchmarking the empirical residual sampling against two alternative constructions: a parametric Gaussian fitted to the residual pool, and a historical bootstrap that draws contiguous residual blocks rather than independent observations.

The study follows the experimental design of Section 4.4.2: the two-stage stochastic optimisation is re-solved under each of the three scenario-construction methods, with the configuration fixed at the balanced objective with CVaR active and the forecasting layer set to walk-forward. The three methods are compared on expected cost, $CVaR_{0.95}$, and service level, and on the cost reduction relative to the deterministic baseline S1 as a comparative diagnostic of how each method performs against the status-quo benchmark.

5.5.1 Empirical versus parametric Gaussian

The parametric Gaussian alternative fits a normal distribution to the walk-forward residual pool of each segment and draws scenarios from the fitted distribution rather than from the empirical pool. The two methods preserve the first two moments of the residual distribution but differ in their treatment of higher-order features: the empirical method preserves the asymmetry, the heavy tails, and any non-Gaussian structure of the realised forecast errors, while the Gaussian method smooths those features away.

The comparison reveals an operationally meaningful gap between the two methods. The empirical method produces a $CVaR_{0.95}$ estimate that is higher than the Gaussian method’s estimate on the same configuration. The mechanism is the heavy-tailed structure of the realised walk-forward residuals: the tail of the empirical pool contains observations that the fitted Gaussian compresses toward the centre of its distribution, and sampling from the Gaussian therefore under-represents the worst-case demand realisations that the CVaR term is designed to bound. An operator who calibrated their risk-management strategy on the Gaussian-derived CVaR would consequently under-provision against tail risk, with the under-provisioning materialising as service-level deterioration on the rare scenarios that the Gaussian dismissed.

The expected cost is closer between the two methods than the CVaR estimate. This is the expected pattern: the mean of the residual distribution is well-approximated by the Gaussian fit, so expected-cost

estimates that integrate over the full distribution converge more easily than tail measures that depend on the upper quantile. The Gaussian method therefore looks adequate when evaluated against the expected-cost KPI alone, and the empirical-versus-Gaussian gap becomes visible only once the tail measure is brought into the comparison.

The operational implication is that the choice of scenario-construction method matters most under tail-risk awareness. Under expected-cost minimisation alone, the framework would produce broadly similar plans under either construction; under CVaR-constrained risk-aware optimisation, the empirical method produces more conservative and operationally faithful plans because it preserves the empirical tail rather than smoothing it.

5.5.2 Empirical versus historical bootstrap

The historical bootstrap alternative draws contiguous blocks of residuals of length $B = 3$ months from the walk-forward pool. The block sampling preserves any month-to-month autocorrelation in the residuals that the independent-sampling construction of Eq. (3.17) discards, and tests whether that dependence has operational consequences on the present case study.

The comparison produces results that are close to the empirical-sampling baseline. Expected cost, CVaR, and service level under the block bootstrap sit within a narrow band around the empirical estimates, with no systematic direction of departure. The interpretation is methodologically informative: residual autocorrelation, while present in the underlying walk-forward residual series at modest values, has limited operational consequence for the single-period optimisation horizon of the present framework. Each scenario in the framework describes a single planning month, and month-to-month dependence is irrelevant to the first-stage decision once the scenario set is fixed; whether the scenarios were drawn independently or as contiguous blocks affects the character of the scenario set marginally but does not substantially shift the operating point that the optimiser selects.

The result confirms the methodological assessment of Section 3.4.2, where the independence-across-time limit of the empirical construction was identified as a mild restriction for single-period optimisation. The empirical near-equivalence between the independent-sampling and block-bootstrap variants supports that assessment and bounds the magnitude of the single-period autocorrelation effect empirically. The result would not generalise to a multi-period rolling-horizon extension, where residual autocorrelation would propagate through successive planning months and the block-bootstrap variant would likely become operationally distinguishable from the independent baseline. This is one of the extensions noted in Chapter 7.

5.5.3 Synthesis: why empirical residual sampling matters

The three-way comparison among empirical, Gaussian, and bootstrap methods supports the framework's methodological commitment to empirical residual sampling on grounds that are operationally distinguishable rather than purely statistical. The empirical method differs from the Gaussian alternative most visibly on the tail measure, where the heavy-tailed structure of the realised forecast errors produces CVaR estimates that the Gaussian fit smooths away. The empirical method differs from the block-bootstrap al-

ternative negligibly on the present single-period framework, because residual autocorrelation has limited operational consequence at that horizon. The methodological argument for the empirical construction therefore rests primarily on the tail fidelity that it preserves relative to parametric alternatives, and secondarily on the generalisability that it offers when the framework is extended to multi-period settings. Both properties motivate the design choice formalised in Eq. (3.17) of Chapter 3.

5.6 Chapter summary

This chapter has reported the empirical findings produced by the framework of Chapter 3 under the experimental conditions of Chapter 4. The results were organised along the operational pipeline: forecasting accuracy first, optimisation outcomes next, and the Pareto frontier together with the Monte Carlo and scenario-construction studies last. The chapter has deliberately concentrated on reporting the empirical outcomes cleanly; the interpretation of these findings against the research questions and hypotheses of Chapter 1 is the content of Chapter 6.

The empirical evidence resolves into a small number of consistent regularities. The forecasting layer produces a clear family-level separation between the tree-based learners with the inverse-MAE ensemble and the three classical baselines: the former are uniformly more accurate across the ten port–container segments under both evaluation protocols, and none of the classical baselines is ever selected by the segment-specific selection rule. Within the dominant family, no single algorithm wins everywhere: CatBoost is the most frequent walk-forward winner but is not the universal choice, and the back-testing and walk-forward protocols disagree on the best algorithm in the majority of segments. The pairwise Diebold–Mariano tests confirm that approximately 80% of the within-segment accuracy comparisons are statistically distinguishable at the 3% level, so the segment-level selection rule acts on genuine predictive differences rather than on sampling artefacts.

The optimisation layer translates the resulting forecasts into repositioning plans whose operational outcomes vary systematically along the three configuration axes that the six end-to-end scenarios exercise. The deterministic baseline (S1) achieves the lowest service level and the highest expected cost, because the shortage penalties dominate its objective; the introduction of machine-learning forecasting and two-stage stochastic optimisation (S2 and S4) reduces expected cost sharply and lifts service level substantially, with the walk-forward configuration reaching 99.5%. The transition from back-testing to walk-forward roughly triples the mean forecast error—a deterioration that a naive expectation would propagate into a proportional increase in operating cost—and yet expected cost barely moves: the gain accrues almost entirely to service-level reliability rather than to nominal first-stage cost. This is the empirical signature of the recourse channel as an absorber of forecast-error volume, identified at the six-scenario level and confirmed at the level of the Pareto frontier, which is near-coincident between the BT and WF pipelines despite the wider WF residual distribution.

The cost–emission Pareto frontier is convex on the case-study network, with a corner-to-corner cost premium of approximately 22% in exchange for an emission reduction of approximately 18% between the cost-optimal and the emission-optimal configurations. The convexity has a direct operational implication for decarbonisation strategy: emission reductions in the neighbourhood of the balanced config-

uration are cheap, while the last percentage points toward the green-optimal corner are expensive, so a green-leaning balanced configuration captures most of the available decarbonisation benefit at a fraction of the cost commitment of the corner itself. The 22% corner-to-corner premium decomposes into a 9% green-routing premium and a 12% CVaR premium, which the pairwise comparisons of Section 5.2.3 isolate and Section 6.4 of Chapter 6 will interpret.

The Monte Carlo robustness analysis confirms that the inter-scenario differences reported in Table 5.6 reflect methodological choices rather than scenario-draw artefacts. The cost ranking, the service-level plateau at 99.5% for the walk-forward configurations, the 18% emission reduction under the green-focus objective, the 9% green-routing premium, and the 12% CVaR premium all sit outside the seed-to-seed variability bands identified in this study, and would be replicated in substantively the same form by an independent analyst working with a different random seed. The scenario-construction-method comparison adds a complementary methodological finding: the empirical residual sampling differs from the Gaussian alternative most visibly on the tail measure, where the heavy-tailed structure of the realised forecast errors produces CVaR estimates that the Gaussian fit smooths away. The empirical method's near-equivalence to the block-bootstrap variant on the present single-period framework bounds the magnitude of the single-period autocorrelation effect empirically and supports the framework's design choice to draw scenarios independently across time.

The chapter that follows develops the mechanistic interpretation of these empirical regularities. The recourse-as-absorber finding, the convex shape of the Pareto frontier, the decomposition of the green-routing and CVaR premia, the three-way separation between the forecasting, recourse, and CVaR layers of uncertainty management, and the boundary conditions under which the framework's empirical results generalise are each addressed in turn, against the research questions and hypotheses formulated in Chapter 1.

Chapter 6

Discussion

This chapter interprets the empirical findings of Chapter 5 in the light of the research questions and hypotheses formulated in Chapter 1. The interpretation is organised thematically rather than along the linear order in which the results were reported. Five mechanistic themes structure the discussion: the absence of a universally dominant forecasting model across heterogeneous port–container segments (Section 6.1), the role of the two-stage recourse channel as an absorber of forecast-error volume (Section 6.2), the operational distinction between bias and variance in what stochastic programming can fix (Section 6.3), the shape of the cost–emission trade-off revealed by the Pareto analysis (Section 6.4), and the three-layer structure of uncertainty management that the framework provides (Section 6.5). The chapter then extracts the practical implications for the principal stakeholder classes in the container logistics ecosystem (Section 6.6) and acknowledges the boundary conditions under which the findings hold (Section 6.7). A short summary closes the chapter in Section 6.8.

The discussion deliberately concentrates on mechanism rather than on additional empirical findings: the role of this chapter is to explain *why* the patterns reported in Chapter 5 take the shape they do, and to translate the resulting understanding into operationally actionable guidance for the stakeholders who would deploy the framework in practice.

6.1 Why no single forecasting model wins everywhere

The most visible regularity in the forecasting results of Section 5.1 is that there is no regularity to speak of at the algorithm level: the best-performing learner shifts across the ten port–container segments and across the two validation protocols. Machine learning models and the inverse-MAE ensemble dominate the three classical baselines uniformly—no classical baseline is ever selected by either protocol—but within that top group the ordering is unstable. The Diebold–Mariano evidence rules out sampling noise as the explanation: with approximately 80% of 280 pairwise tests rejecting equal predictive accuracy at the 3% level, the family-level gap between tree-based learners and classical baselines is statistically robust, and the within-family ordering is not driven by random variation in any single test window. This bears directly on Research Question RQ1 of Section 1.3, which asked whether the ranking of forecasting approaches is stable under operationally realistic out-of-sample evaluation; the empirical answer is that

it is not, and that the instability is itself a finding of practical significance.

The heterogeneity is substantive rather than incidental. Ports have different operational behaviours: a chronic transshipment hub like Gioia Tauro absorbs trade-imbalance flows that behave differently from the export-driven cycles of La Spezia or Trieste, and the demand series at the two surplus ports are themselves structurally distinct from each other—Genova’s flow is driven by Northern Italian manufacturing exposure, while Gioia Tauro’s flow is driven by the global deep-sea network in which it acts as a Mediterranean hub. Demand patterns also differ between 20-foot dry van and 40-foot high-cube containers within the same port, reflecting different cargo mixes and different downstream depot economics. A forecasting layer that pools across these heterogeneous segments sacrifices segment-level predictive performance in segments where another algorithm would have performed better, and the empirical evidence of Table 5.2 shows that those segments are the majority on the present case study. This finding is consistent with the broader machine-learning literature on heterogeneous data, where no-free-lunch results have been repeatedly identified across application domains (Xu et al., 2022; Chan et al., 2019; Henesey and Liu, 2020); the present study confirms the no-free-lunch pattern empirically in the empty-container repositioning context, where it has received limited attention to date, and confirms Hypothesis H1 of Section 1.4.

The choice of validation protocol matters for deployment in a way that the no-free-lunch observation alone does not capture. Back-testing evaluates a model on a single held-out window and assumes the operating regime stays put across that window; walk-forward retrain at every fold and asks the model to forecast under conditions that resemble the information state at deployment. The two protocols disagree on the specific learner in eight of the ten segments, which means that a system calibrated on back-testing alone would go live with a forecaster that, on the majority of its segments, has never been tested under its operating regime. This is more than a statistical quibble: an operator deploying the framework with back-testing-only calibration would systematically under-test the algorithm’s behaviour during the disrupted training window of 2020–2022, because the calendar-2025 back-testing window does not contain those folds. The walk-forward protocol therefore drives the rest of the pipeline by design (Section 3.4.2), not because it is the more accurate protocol on average—in fact, walk-forward errors are larger than back-testing errors on the same segments—but because it is the protocol that exposes the algorithm to the deployment regime (Zugno et al., 2025).

6.2 The recourse channel as an error absorber

The most counterintuitive finding in Chapter 5 is also the one with the clearest mechanism behind it. The walk-forward protocol roughly triples average forecast error relative to back-testing across the ten segments, and a naive expectation would project that the threefold deterioration in forecast accuracy must propagate into a comparable deterioration in operating cost. It does not. The pairwise comparison between scenarios S2 (back-testing) and S4 (walk-forward) in Table 5.6 shows that expected cost moves by less than one percent between the two protocols, even as service-level reliability rises from approximately 97% to 99.5%. The near-coincidence of the BT and WF Pareto frontiers (Section 5.3.3) extends this finding from a single configuration to the full cost–emission frontier: the two pipelines identify

essentially the same Pareto-efficient operating points despite the substantial gap in underlying forecast accuracy. This empirical pattern is the direct test of Research Question RQ2 and confirms Hypothesis H2 of Section 1.4: the operational return on better forecasting accrues primarily to service-level reliability rather than to nominal first-stage cost.

The mechanism is structural. Two-stage stochastic programming commits the first-stage plan before demand is observed and then absorbs the realised imbalance, scenario by scenario, through a recourse channel priced at the cheapest emergency routing available to each destination port. When the empirical residual distribution widens—as it does when moving from back-testing to walk-forward—the optimiser does not respond by inflating the first-stage commitment to insure against the wider range of demand realisations. It responds by reshaping the scenario-specific second-stage flows: each scenario draws its own recourse pattern from the under-shipment and over-shipment variables $y_j^{+, (s)}$ and $y_j^{-, (s)}$ of Eq. (3.26)–(3.27), and the aggregate effect across the $S = 30$ scenarios absorbs the additional uncertainty without inflating the first-stage expense. The cost of being wrong about demand is paid in operational adjustment, not in upfront expenditure.

The operational consequence is sharp. What changes between the back-testing and walk-forward pipelines is not the cost of the plan but the rate at which the plan delivers on its service commitment. Service level rises from 96.8% under back-testing (scenario S2) to 99.5% under walk-forward (scenario S4), and the same plateau holds for all three walk-forward configurations regardless of the objective weighting or the CVaR setting (scenarios S4, S5, S6 in Table 5.6). Accuracy in the forecasting layer buys reliability first and cost second; under the cost structure of the present network, it buys almost entirely reliability. The implication for the framework’s deployment value is contingent on the contractual environment in which the framework is deployed. For a carrier whose service-level commitment carries thin penalty clauses, the return on better forecasting is correspondingly thin: the realised cost savings are small, and the reliability gain has no direct monetary counterpart. For a carrier whose service-level commitment carries substantial penalty clauses—missed deliveries triggering contractual liquidated damages, for example, or service-level agreements with key customers—the reliability gain converts directly into avoided penalties, and the return on better forecasting is substantial. The same framework therefore produces materially different deployment cases for different operators, and the empirical results of Chapter 5 should be read as quantifying the reliability gain rather than as delivering a single headline return figure.

The same mechanism also explains the near-coincidence of the BT and WF Pareto frontiers identified in Section 5.3.3. The cost–emission trade-off is governed by the network topology, the modal economics, and the capacity profile of the three transport modes, not by the forecasting accuracy. As long as the recourse channel absorbs the forecast-error volume into second-stage flows rather than first-stage commitment, the Pareto frontier reflects the underlying physical and economic structure of the network, and the forecasting layer only modulates the service-level reliability that the chosen operating point delivers. The result is robust within the structural balance of the case-study network and is one of the central operational findings of the framework. Whether it generalises to networks with materially different recourse economics is one of the boundary conditions addressed in Section 6.7.

6.3 Bias, variance, and what stochastic programming can fix

The recourse-as-absorber mechanism of Section 6.2 works on the variance component of forecast error, not on its bias. This distinction is rarely made explicit in the empty container repositioning literature, yet it has direct operational consequences for the allocation of forecasting effort under stochastic optimisation and deserves a separate discussion.

6.3.1 The distinction in operational terms

A forecasting algorithm whose errors are unbiased but spread produces a residual distribution centred on zero with a non-trivial dispersion: positive and negative misses are roughly balanced in expectation, and the magnitude of any individual miss varies across the test window. This is the residual profile that the framework’s empirical scenario construction (Section 3.4.2) is designed to exploit. The two-stage stochastic programme commits a first-stage plan calibrated to the central tendency of the residual distribution and lets the recourse channel absorb the realised deviation; the under-shipment variable $y_j^{+, (s)}$ activates on scenarios where demand overshoots the forecast, and the over-shipment variable $y_j^{-, (s)}$ activates on scenarios where demand undershoots. The unbiased-but-spread residual profile produces both types of scenarios with approximately equal frequency, and the framework absorbs each through its corresponding recourse variable.

A forecasting algorithm whose errors are biased, by contrast, produces a residual distribution whose mean is offset from zero: every scenario shifts in the same direction relative to the realised demand. Stochastic programming offers little advantage over deterministic planning in this regime. If every scenario systematically over-states demand, the optimiser pre-commits a first-stage plan that over-supplies every destination, and the recourse channel activates almost exclusively on the over-shipment side, accumulating storage costs c^- across scenarios. If every scenario systematically under-states demand, the symmetric problem arises: under-shipment costs c_j^+ accumulate, and the framework absorbs the bias as a chronic shortage rather than absorbing it operationally. In both cases the recourse channel does its mechanical job—it covers the realised imbalance scenario by scenario—but it cannot correct the underlying bias of the forecasting layer. The first-stage plan inherits the bias from the forecast and pays for it through one-sided recourse activity.

6.3.2 Implications for forecasting effort

Bias correction and variance reduction look like complements in a forecasting-accuracy table, but they are substitutes in operational terms. Two forecasting algorithms with identical MAE on the same segment may carry very different operational implications under stochastic optimisation: a high-bias, low-variance algorithm and a low-bias, high-variance algorithm achieve the same MAE through completely different error structures, and the recourse channel responds to them in qualitatively different ways. The high-bias algorithm produces a one-sided recourse profile that no stochastic optimisation layer can fix; the low-bias algorithm produces a balanced recourse profile that the stochastic optimisation layer absorbs as designed. An accuracy-only evaluation of the forecasting layer would treat the two algorithms

as interchangeable, but they are not.

The operational implication for the deployment of the framework is that forecasting effort should be allocated primarily to bias reduction, not to variance reduction. The recourse channel handles variance for free, in the operational sense that it neither inflates first-stage cost nor erodes service-level reliability under wider residual distributions, as Section 6.2 established. It does not handle bias for free: every unit of bias in the forecasting layer translates into a proportional unit of one-sided recourse activity that the optimisation layer cannot avoid. A forecasting investment that reduces residual variance without addressing residual bias buys service-level reliability at no operational cost; a forecasting investment that addresses residual bias buys both cost reduction and reliability simultaneously. The framework’s segment-specific selection rule of Eq. (3.11) acts on MAE rather than on a bias-corrected metric, which means that it may select algorithms whose lower MAE arises through variance reduction at the expense of bias—a phenomenon worth identifying explicitly even if it does not change the algorithmic selection on the present case study, where the walk-forward residuals of the tree-based learners are approximately mean-zero by construction.

6.3.3 Why this distinction matters for the literature

The bias–variance distinction has been extensively developed in the machine-learning literature as a property of estimators (Hastie et al., 2009), but its operational consequences for stochastic logistics optimisation have received less systematic attention. The distinction matters here because it changes the interpretation of the forecasting accuracy results in Chapter 5. The roughly threefold increase in walk-forward MAE relative to back-testing is, in the present framework, almost entirely a variance increase: the walk-forward residuals span a wider range than the back-testing residuals while remaining approximately centred on zero. The recourse-as-absorber mechanism works precisely because this variance increase carries no bias component, and the framework’s operational outcomes are robust to it. A different operating environment—a network with persistent demand bias, or a forecasting setup whose residuals exhibit systematic drift over the deployment horizon—would activate a different operational regime in which the recourse channel cannot substitute for forecasting quality. The recourse-as-absorber finding is therefore conditional on the bias profile of the forecasting layer, and the deployment context should be assessed against this condition before relying on the mechanism for operational planning.

6.4 The shape of the cost–emission trade-off

The Pareto frontier reported in Section 5.3 is convex on the case-study network and exhibits a corner-to-corner cost premium of approximately 22% in exchange for an emission reduction of approximately 18%. This section interprets the shape and the magnitude of the trade-off against Research Question RQ3 of Section 1.3, decomposes the 22% premium into its green-routing and CVaR components, and extracts the practical implication for decarbonisation strategy.

6.4.1 The convexity of the frontier

The convexity of the frontier means that the marginal cost of emission reduction is not constant along the curve. Section 5.3.2 identified three regions, each with a distinct operational character. Near the cost-optimal corner, emission reductions are cheap, obtained by shifting short corridors from road to rail wherever rail capacity has not yet been exhausted; these substitutions trade a small road cost for a substantial emission saving on each TEU shifted, producing a shallow initial slope. In the middle region, the available short-corridor rail capacity is progressively exhausted, and further emission reductions require either rail-to-sea substitutions on medium corridors or displacement of road volume onto longer short-sea legs that carry a cost premium without a proportional emission saving. Near the emission-optimal corner, the last percentage points of emission reduction are bought at a marginal cost that substantially exceeds the average cost across the frontier, because the network has exhausted its cheapest decarbonisation levers and the remaining substitutions trade large cost increases for small emission gains. The convexity is therefore not a stylised assumption; it is the visible imprint of the network's modal economics and capacity constraints on the trade-off space, and it is consistent with the broader intermodal-logistics literature on cost–emission trade-offs (Sharma and Gupta, 2017; Holguín-Veras et al., 2021; Octaviansyah et al., 2024).

6.4.2 Decomposition of the corner-to-corner premium

The corner-to-corner cost premium reported in Section 5.3.1 reflects the combined effect of two mechanisms: the cost of shifting toward cleaner transport modes and the cost of controlling tail-risk through the CVaR term. The first component can be read most directly from the comparison between scenarios S5 and S6. Both scenarios use the walk-forward forecasting pipeline, the two-stage stochastic optimisation model, and CVaR risk control; they differ primarily in the objective configuration. Scenario S5 adopts the green-focus configuration, whereas scenario S6 adopts the cost-focus configuration. The cost gap between the two therefore provides the clearest estimate, within the six-scenario design, of the green-routing premium, which is approximately 9%. The CVaR-related premium is more difficult to isolate within the six end-to-end scenarios because the scenario that removes CVaR under the walk-forward pipeline, S4, uses the balanced objective configuration, whereas the CVaR-enabled cost-focused scenario, S6, uses the cost-focus configuration. The comparison between S4 and S6 therefore should not be interpreted as a clean single-effect estimate of the CVaR premium. It is better interpreted as a cost-aware risk-control comparison, in which the introduction of CVaR is accompanied by a change in objective weights. A clean CVaR premium would require an additional fixed-objective diagnostic in which the same forecasting protocol and the same objective weights are solved once with CVaR and once without CVaR. Within this limitation, the results still show that tail-risk control carries a non-negligible robustness cost. The increase in expected cost observed when moving from the unconstrained walk-forward balanced configuration to the CVaR-enabled cost-focused configuration is therefore evidence of a risk-control premium, but not a pure CVaR premium in the strict experimental sense. For this reason, the reported cost increase should be discussed as an indicative robustness premium rather than as a cleanly isolated CVaR effect.

6.4.3 Practical implication: a green-leaning balanced configuration

The convexity of the frontier has a direct operational implication for decarbonisation strategy. The cheapest emission reductions are not at the green-optimal corner but in its neighbourhood, and an operator targeting an emission reduction in the 10–15% range relative to the cost-optimal benchmark can achieve that target at a cost premium of approximately 5–10%, well below the 22% corner-to-corner premium. The marginal cost of further emission reductions beyond that range rises sharply, so a green-leaning balanced configuration captures the bulk of the available decarbonisation benefit at a fraction of the cost commitment of the corner itself.

The implication generalises a finding that has been observed in adjacent transport-decarbonisation contexts (Sharma and Gupta, 2017; Holguín-Veras et al., 2021) but that the empty-container repositioning literature has rarely articulated in explicit Pareto terms. The operational decision facing a port operator or shipping line under emission-reduction pressure is not whether to adopt the cost-optimal or the green-optimal configuration: the binary choice forfeits most of the available benefit on one side and pays a disproportionate premium on the other. The decision is where to position along the convex frontier, and the framework’s contribution on this dimension is to make the trade-off visible so that the position can be chosen explicitly rather than inherited from a single-objective optimiser. The third research question of Section 1.3, which asked about the achievable trade-offs between cost, service level, and CO₂ emissions under multimodal optimisation, is therefore answered in the form of a convex frontier whose shape can be exploited by the planner rather than as a single quantitative trade-off ratio, confirming Hypothesis H3 of Section 1.4.

6.5 Three layers of uncertainty, three different jobs

The framework treats demand uncertainty at three points along the operational pipeline, and each point does an operationally distinct job. Machine learning forecasting reduces the average forecast error and improves the central tendency of the demand estimate. Two-stage recourse absorbs the realised forecast error scenario by scenario, reshaping second-stage flows to satisfy whatever imbalance materialises once demand is observed. The CVaR constraint bounds the tail of the cost distribution by penalising the worst-5% outcomes in the objective function. This section interprets the three-layer structure against the empirical evidence of Chapter 5 and against Research Question RQ3 of Section 1.3, and identifies the non-substitutability between the three layers as a central design property of the framework.

6.5.1 What each layer does

The forecasting layer is the source of the central demand estimate that the optimiser uses to fix the first-stage plan. Reducing average forecast error through better algorithms or more demanding validation protocols shifts that central estimate closer to the realised demand, but the empirical evidence of Section 5.2.3 documents that the shift translates almost entirely into service-level reliability rather than into first-stage cost. The forecasting layer therefore governs the *accuracy* of the central plan, in the operationally narrow sense that the layer determines how often the realised demand is close to the planned

inbound flow.

The recourse layer absorbs whatever residual imbalance remains once demand is realised. The under-shipment variable $y_j^{+, (s)}$ and the over-shipment variable $y_j^{-, (s)}$ of Section 3.8.4 handle this absorption scenario by scenario, with the cost of recourse priced at the route-aware penalty c_j^+ for shortages and the storage cost c^- for excess inventory. The pairwise comparison between back-testing and walk-forward scenarios (S2–versus–S4 and S3–versus–S5) documented in Section 5.2.3 demonstrates that the recourse layer behaves as designed: as the empirical residual distribution widens, the layer reshapes second-stage flows without inflating first-stage cost. The recourse layer therefore governs the *operational adjustment* of the plan to realised conditions, in the sense that the layer determines how the optimiser pays for being wrong about demand once demand is observed.

The CVaR constraint bounds the right tail of the cost distribution. The Rockafellar–Uryasev linearisation embedded in Constraint (3.35) penalises the worst tail of the scenario-cost distribution at a rate governed by the CVaR confidence level α and the risk-aversion weight λ . This steers the optimiser away from plans whose expected cost may be attractive but whose tail-cost exposure is operationally undesirable.

In the six end-to-end scenarios, the role of CVaR is visible most clearly through the behaviour of the CVaR-enabled configurations, but its pure marginal effect is not fully isolated by the S4–S6 comparison because the objective configuration also changes from balanced to cost-focused. The S4–S6 comparison should therefore be interpreted as a cost-aware risk-control comparison rather than as a clean CVaR-only experiment. A clean isolation of the CVaR effect would require solving the same walk-forward stochastic configuration with identical objective weights, once with CVaR and once without CVaR. Subject to this experimental limitation, the results indicate that CVaR introduces a robustness premium by reducing exposure to extreme demand realisations at the cost of higher expected cost.

6.5.2 Why the three layers are non-substitutable

The three layers are non-substitutable in the sense that none of them can perform the operational job of any of the others. Recourse without CVaR keeps expected cost low but offers no commitment about the worst-case scenario: the optimiser is free to accept plans whose tail-cost is arbitrarily bad as long as the probability-weighted average across scenarios is small. The empirical evidence is the gap between the $\text{CVaR}_{0.95}$ measures of the unconstrained balanced configurations (S2 and S4) and the constrained variants (S3, S5, S6): under the unconstrained configurations the tail-cost is not reported because it is not a decision variable in the objective, but the underlying scenario-cost distribution is wider than under the constrained variants. Recourse handles the average miss but does not bound the worst-case miss.

CVaR without recourse imposes a tail-cost bound on a deterministic plan that has no mechanism to comply with it. A deterministic linear programme in the form of scenario S1 commits a single first-stage plan calibrated to a point forecast, and the resulting tail-cost is governed by the gap between the point forecast and the realised demand. Adding a CVaR constraint to such a formulation would force the optimiser to inflate the first-stage commitment to cover the worst-case scenarios in advance, which contradicts the operational logic of the deterministic formulation; the constraint would tighten the bound

at the cost of the central plan's attractiveness. The two-stage stochastic structure is what makes the CVaR constraint operationally sensible: the optimiser can leave the first-stage plan close to its expected-cost optimum and let the recourse channel absorb the tail through scenario-specific adjustment, with the CVaR constraint pricing the resulting tail-cost rather than inflating the first-stage commitment.

Better forecasting cannot substitute for either the recourse layer or the CVaR constraint. A forecaster with vanishingly small error would indeed obviate the need for both layers in the idealised limit, but no forecaster on realistic demand series achieves vanishingly small error, and the empirical evidence of Chapter 5 shows that even the best walk-forward configuration on the present case study leaves a residual MAE on the order of hundreds of TEU per segment. The recourse layer exists to absorb that residual, and the CVaR constraint exists to bound the tail of the resulting cost distribution; neither can be substituted by an upstream improvement in forecasting accuracy under realistic operating conditions.

6.5.3 When each layer is most valuable

The three-layer structure also clarifies when each layer delivers its largest operational value. The forecasting layer delivers most under contractual environments where service-level commitments carry material penalties: the reliability gain documented in Section 6.2 converts into avoided penalties most directly when missed deliveries trigger liquidated damages or service-level-agreement breaches. The recourse layer delivers most under operational conditions where demand uncertainty is wide: the broader the realised residual distribution, the more operational benefit the recourse channel extracts through second-stage reshaping. The CVaR constraint delivers most under high-impact disruptions: a new pandemic wave, a geopolitical disruption that reroutes major trade lanes, or a sudden supply-side shock all move demand realisations into precisely the tail region that the constraint is designed to bound, and the robustness premium that the constraint charges buys distributional protection rather than expected savings. The operational relevance of the CVaR constraint therefore grows sharply under high-impact disruptions, which is the regime in which the framework's empirical advantage over the deterministic baseline is most pronounced.

6.6 Managerial and policy implications

The framework's empirical contributions and the mechanistic interpretation developed in the preceding sections translate into operationally distinct implications for the four principal stakeholder classes in the container-logistics ecosystem: shipping lines, port operators, inland transport providers, and regulators. This section extracts those implications stakeholder by stakeholder, keeping the framework's findings firmly anchored to the empirical evidence of Chapter 5 rather than extrapolating beyond the case-study network.

6.6.1 Shipping lines

Shipping lines are the primary owners of the container fleet and the principal beneficiaries of the framework's repositioning recommendations. Three implications follow from the empirical findings.

First, the dominant operational return from the framework's adoption is service-level reliability rather than nominal cost reduction. The walk-forward configurations of Table 5.6 deliver a service level of 99.5% against the deterministic baseline's 94.1%, with the cost column moving only modestly across the stochastic configurations relative to the deterministic benchmark. For shipping lines whose contracts with key customers carry liquidated-damages clauses on missed-delivery thresholds, this reliability gain converts directly into avoided penalties; for lines whose contracts carry thinner penalty clauses, the framework's benefit accrues primarily to customer-relationship quality and to the reduced operational stress of running close to the service-level boundary. The deployment case for the framework should therefore be built around contractual exposure rather than around cost reduction alone.

Second, the forecasting layer that drives the framework should be evaluated on bias rather than on variance, as Section 6.3 argued. A forecasting investment that addresses bias delivers both cost reduction and reliability improvement; a forecasting investment that addresses variance alone delivers reliability improvement at zero nominal cost saving. Shipping lines investing in predictive analytics for repositioning should audit their forecasting layer for bias and prioritise bias-reducing model changes over generic accuracy improvements.

Third, the framework supports a green-leaning balanced configuration as the operationally attractive default, as Section 6.4.3 identified. Shipping lines facing emission-reporting pressure under the European Union's Emissions Trading System inclusion of maritime transport or the International Maritime Organization's greenhouse-gas strategy can capture the bulk of the available decarbonisation benefit at modest cost premium without committing to the emission-optimal corner. The framework makes the trade-off visible at the planning interface, which supports an explicit positioning decision rather than a tacit one.

6.6.2 Port operators

Port operators do not own the container fleet, but they bear the operational consequences of imbalances at the yard and the throughput implications of inbound and outbound repositioning flows. Three implications follow.

First, the framework provides port operators with explicit visibility into the inbound repositioning volume that each carrier's plan implies for their terminal. The shipment-details panel of the results tab (Section 3.9.5) exposes the origin–destination flows by mode, and a port operator who runs the framework against their own forecast of customer demand can anticipate the inbound volume that the repositioning network will deliver in the planning month. This visibility supports yard-capacity planning, gate-operations scheduling, and the coordination with inland transport providers that the inbound volume will require.

Second, the green-focused configuration of the framework typically shifts repositioning flows away from the road mode and toward rail and short-sea shipping. For port operators with established rail-terminal infrastructure or with short-sea connectivity to the hinterland, this modal-shift trend translates into higher intermodal throughput and stronger justification for further rail-terminal investment. For port operators without such infrastructure, the trend identifies a strategic gap: as carriers adopt green-leaning

planning frameworks, ports without intermodal capacity will lose repositioning business to ports that have it.

Third, the framework's CVaR constraint protects the network against extreme demand realisations that would otherwise produce sharp yard-occupancy spikes at deficit ports or persistent over-stocking at surplus ports. The robustness premium that the constraint charges is paid by the shipping line in expected cost; the operational benefit accrues jointly to the line (through avoided service-level breaches) and to the port operator (through smoother yard occupancy). The framework therefore aligns the incentives of carriers and ports on robustness more closely than a cost-only deterministic framework would.

6.6.3 Inland transport providers

Inland transport providers—road hauliers, rail operators, and short-sea shipping lines—constitute the operational layer through which the framework's repositioning plans are executed. Two implications follow.

First, the framework's modal-mix outcomes documented in Section 5.2.2 show a substantial rail and short-sea share under the green-focus configuration, with road absorbing the residual volume that the cleaner modes cannot accommodate within their capacity constraints. Inland transport providers serving the Italian Mediterranean corridors should anticipate the demand-side pull this trend implies for their respective modes: rail operators on the medium corridors of the network and short-sea operators on the longer routes into Gioia Tauro stand to gain volume from the green-leaning trend, while road hauliers on the same corridors stand to absorb the overflow when the cleaner modes exhaust their capacity. The capacity-constraint pattern identified in Section 5.2.2 is therefore itself an investment signal for the inland transport sector.

Second, the route-aware recourse penalty of Eq. (3.23) prices emergency repositioning at the cheapest reachable network arc plus a service-delay surcharge. The operational implication for inland transport providers is that emergency-recourse demand for their services is priced at the marginal cost of the cheapest reachable corridor plus the delay penalty, which sets a natural ceiling on the premium that a provider can charge for short-notice repositioning services. Providers seeking to capture emergency-recourse business should position themselves on the cheapest reachable corridors to deficit ports, since those are the corridors that the framework's recourse penalty calibrates against.

6.6.4 Regulators and policy-makers

Regulators and policy-makers stand outside the operational flows of the framework but shape the incentive environment in which the operational stakeholders deploy it. Three implications follow.

First, the framework demonstrates that multimodal repositioning across road, rail, and short-sea shipping can achieve substantial emission reductions ($\approx 18\%$ in the green-focused configuration relative to the balanced reference) at modest cost premium ($\approx 9\%$ green-routing premium plus a $\approx 12\%$ CVaR premium when tail-risk is controlled). This evidence supports the broader policy argument that modal-shift incentives—rail infrastructure investment, short-sea connectivity support, road-pricing schemes that internalise the emission externality—deliver decarbonisation returns when carriers face the trade-off ex-

plicitly. Modal-shift incentives that act only on the supply side (rail capacity expansion without a corresponding shift in the demand-side incentives) would risk producing infrastructure under-use; modal-shift incentives that act jointly on both sides (capacity expansion plus a transparent price signal that operators can incorporate into their planning frameworks) would not.

Second, the framework's emission-reduction outcomes are governed by the objective weighting rather than by the optimisation paradigm or the forecasting protocol, as Section 5.2.3 identified. The implication for regulatory design is that emission-reporting requirements should focus on the operational outcome (the realised modal mix, the realised emission footprint per TEU repositioned) rather than on the input configurations that produced it. Two operators using the same framework under different objective weightings will report materially different emission outcomes; two operators using different frameworks under the same objective weighting will report similar emission outcomes. The regulatory focus on outcomes rather than on methods is the appropriate design choice for this evidence base.

Third, the CVaR-constrained configurations deliver the framework's largest robustness contribution under high-impact disruptions, as Section 6.5.3 identified. Policy responses to maritime supply-chain disruptions—pandemic-era port-congestion crises, geopolitical re-routing events, and similar—would benefit from regulatory recognition that risk-aware planning frameworks deliver supply-chain resilience that expected-cost minimisation does not. Regulatory guidance that encourages the adoption of risk-aware planning, or that conditions public-sector contingency support on the use of such frameworks, would align private operational incentives with the public interest in supply-chain resilience.

6.7 Boundary conditions and deployment considerations

The empirical findings of Chapter 5 and the mechanistic interpretations developed in Sections 6.1–6.6 hold under a set of structural conditions that the case-study network satisfies and that a deployment context should be assessed against before relying on the framework operationally. This section makes those conditions explicit. The intent is not to disqualify the framework outside its tested envelope, but to identify where its mechanistic findings depend on the case-study structure and where they would generalise without qualification.

6.7.1 The calibration boundary

The dataset used throughout this thesis is constructed through the calibrated procedure documented in Section 3.2.4, anchored to publicly available aggregate statistics from Assoparti and Eurostat rather than to raw carrier-level operational records. This choice is established practice in stochastic logistics research when carrier-level records are inaccessible (Song and Carter, 2009; Vassos et al., 2025; Sommer et al., 2025), and it preserves the structural imbalances, seasonality, and disruption signatures of the underlying flows. The methodological contributions of the thesis—the forecast–optimisation coupling, the empirical scenario construction, the route-aware recourse penalty, the dual-branch architecture—do not depend on the precise absolute values of the underlying data and would operate on proprietary carrier records under the same mathematical formulation. The relative comparisons across configurations, including the 9%

green-routing premium, the 12% CVaR premium, and the 18% emission reduction under the green-focus configuration, are structural findings of the framework on the case-study network and are similarly robust.

Absolute cost and emission figures, by contrast, should be interpreted conditional on the calibration. The headline EUR figures of Table 5.6 and the absolute tonnage figures elsewhere in the chapter reflect the calibrated arc parameters of Section 3.7 rather than direct measurement of carrier operations, and an operator running the framework with proprietary data would obtain different absolute figures even though the relative pattern across configurations would remain the same. The framework's value as a decision-support tool lies in the relative comparisons it enables rather than in the absolute forecasts it produces; the absolute figures reported here serve to illustrate the scale of the operational outcomes rather than to deliver a standalone forecast of EUR cost in deployment.

The natural extension of this boundary is validation on proprietary carrier data. The framework is structurally ready for such validation: the data-ingest tab of the platform (Section 3.2.6) accepts the same schema regardless of whether the input is the calibrated dataset of this thesis or a proprietary operational dataset, and the subsequent steps of the pipeline depend on the schema rather than on the data source. Proprietary validation is the single most consequential extension for the framework's deployment readiness and is discussed as a future-work direction in Chapter 7.

6.7.2 The single-period horizon

The framework is formulated as a single-period two-stage stochastic programme. The opening stock of empty containers is treated as a known input at the start of the planning month, and the recourse decisions concern only the redistribution of that stock across the network within the planning month. Inventory carry-over between consecutive planning months is not modelled: a container that arrives at a destination port in the planning month under one realisation of demand does not appear as an updated supply input in the following month's planning instance. The framework therefore captures the operational decision a port planner faces at the start of each month, but it does not optimise across months as a sequence.

This boundary has two consequences worth naming. First, the residual autocorrelation that the empirical scenario sampling discards (Section 3.4.2) is operationally negligible at the single-period horizon, as the scenario-construction-method comparison of Section 5.5.2 confirmed empirically. A multi-period rolling-horizon extension would change this calculus: residual autocorrelation would propagate through successive planning months, and a block-bootstrap scenario construction would likely become operationally distinguishable from the independent-sampling baseline. The methodological choice to sample residuals independently across time is therefore conditional on the single-period scope.

Second, the value of the stochastic optimisation relative to a deterministic baseline is bounded by what the recourse channel can absorb within a single planning month. A multi-period formulation would allow the recourse mechanism to extend across months: a shortage at a destination port in one month could be partly mitigated by adjusting the inbound flow in the following month, and the optimiser would have an additional degree of freedom that the present formulation does not expose. The framework's operational outcomes would likely improve under such an extension, but the empirical magnitude of the improvement is an open question that proprietary multi-period data could resolve. The rolling-horizon

extension is the second most consequential future-work direction, after proprietary validation.

6.7.3 Cross-port independence in scenario construction

The empirical scenarios of Eq. (3.17) sample residuals independently across ports, which is a more substantive simplification than the independence across time discussed in Section 6.7.2. During a system-wide shock such as the COVID-19 disruption, forecast errors tend to be positively correlated across the network: when demand overshoots at Genova, it is likely to overshoot at La Spezia and at Trieste as well, because the underlying disruption affects all five ports through shared exposure to global supply-chain conditions. Independent draws across ports under-represent this co-movement and therefore underestimate the tail-risk that the CVaR constraint is designed to bound.

The operational consequence is that the $\text{CVaR}_{0.95}$ measures reported in Table 5.6 are conservative-on-the-low-side estimates of the true tail-risk under realistic correlated demand shocks. The recourse-as-absorber mechanism of Section 6.2 is similarly optimistic under cross-port co-movement: a synchronised demand overshoot across multiple ports would exhaust the available recourse capacity faster than independent overshoots would, because the recourse channel itself draws on the same network of arcs whose capacity is fixed by the parameterisation of Section 3.7. The framework's operational outcomes under correlated demand shocks would therefore be less favourable than the empirical results suggest, with the size of the deterioration governed by the strength of the cross-port residual correlation that the independent-sampling construction sets to zero.

A multivariate residual model that preserves the empirical cross-port covariance is the natural extension that would address this boundary. The construction would replace the per-segment residual pool of Eq. (3.16) with a joint residual sample that captures the contemporaneous correlation across ports, and the scenario construction of Eq. (3.17) would draw from the joint sample rather than from independent per-segment samples. The tail-risk estimates produced under this construction would tighten relative to the independent baseline, and the CVaR constraint would deliver more accurate protection against system-wide shocks. This is the third future-work direction discussed in Chapter 7.

6.7.4 Network specifics and generalisation

The case-study network consists of five Italian ports along the Western Mediterranean and Northern Adriatic corridors, connected by a multimodal network whose arc parameters are calibrated to documented monthly throughput levels on Italian logistics corridors. The specific results of Chapter 5 should not be assumed to generalise mechanically to networks with different structural characteristics, regulatory environments, or modal mixes. Three structural features of the Italian network shape the framework's empirical outcomes in ways that may not transfer unchanged.

First, the surplus–deficit balance of the network is structurally moderate. Genova and Gioia Tauro carry a combined surplus of approximately 5,486 TEU per planning month against a combined deficit of approximately 3,079 TEU at the three deficit ports, giving a network-wide surplus of approximately 2,407 TEU. The network is therefore physically feasible by a comfortable margin, and the optimisation challenge is one of geographic reallocation rather than aggregate scarcity. Networks with tighter aggre-

gate balance, or with persistent network-wide shortage, would exercise the framework under materially different operational conditions: the recourse channel would become more expensive on average as c_j^+ activates more frequently, and the CVaR constraint would bind more aggressively because the right tail of the cost distribution would carry more shortage cost.

Second, the modal economics of the Italian network favour rail and short-sea over road on the medium and long corridors. Networks with different modal-cost relationships—North American networks with stronger road dominance, for example, or Northern European networks with more capacity-constrained rail—would produce different Pareto frontiers and different modal-mix outcomes under the green-focus configuration. The convexity of the Pareto frontier is a structural property of multimodal networks with heterogeneous mode costs, but the specific slope and the corner-to-corner premium depend on the relative modal costs of the network in question.

Third, the planning horizon is monthly, which aligns with the planning cadence of many shipping lines and depot operators but does not capture intra-month dynamics such as vessel-call clustering or daily yard congestion. Networks operating under tighter planning horizons would face the framework's decisions in a different temporal context, and the operational outcomes would respond to that context in ways that the present formulation cannot directly predict.

The framework's mechanistic findings—the recourse-as-absorber mechanism, the convex Pareto frontier, the three-layer separation of uncertainty management—are structural properties of the methodological design rather than artefacts of the case-study network, and would be expected to hold qualitatively across networks with similar multimodal structure. The quantitative magnitudes reported in Chapter 5, by contrast, are case-study-specific and should be re-estimated on each new deployment context before being used in planning decisions.

6.7.5 Live deployment beyond the framework itself

Live deployment of the framework in an operational planning environment is constrained less by the framework itself than by the data access required to feed it. Monthly port-level flows disaggregated by container type sit across carrier, terminal-operator, and customs systems that are rarely integrated, and most cross-system access is mediated by bilateral data-sharing agreements between specific parties (United Nations Conference on Trade and Development, 2023). The data-access barrier is the binding constraint on the framework's deployment readiness, not the computational or methodological complexity of the framework itself: the timing profile shows that the full pipeline runs in well under five minutes on commodity hardware, comfortably within the planning cadence of a monthly operational cycle, so the framework's computational footprint is not a deployment-readiness limitation.

The outputs of the framework—the repositioning plan, the service-level forecast, the cost–emission trade-off curve, the modal-mix decomposition—are designed to be surfaced through existing ERP or terminal-operating-system interfaces as a recommendation module rather than as a stand-alone tool. The architectural choice supports integration with existing carrier or port systems without forcing a wholesale replacement of those systems, which is the realistic deployment pattern for a research-grade prototype operating against production infrastructure. The CVaR confidence level α and the risk-aversion weight

λ remain exposed at the configuration interface so that operators with different risk appetites can run the same pipeline under their own settings without modifying the underlying code.

6.8 Chapter summary

This chapter has interpreted the empirical findings of Chapter 5 along five mechanistic themes and extracted their implications for the principal stakeholder classes in the container-logistics ecosystem, within the boundary conditions that the case-study network satisfies.

The forecasting layer admits no universally dominant algorithm across the ten port–container segments, and the back-testing and walk-forward protocols disagree on the best algorithm in the majority of segments. The heterogeneity is operational rather than incidental, and the walk-forward protocol is chosen by design as the operationally honest basis for downstream planning, not because it is the more accurate protocol on average. The recourse layer absorbs the resulting forecast-error volume into reshaped second-stage flows scenario by scenario, so that improvements in forecasting quality translate primarily into service-level reliability rather than into proportional reductions in nominal first-stage cost. This is the central operational finding of the framework, and it has the testable implication that the deployment case for the framework should be built around contractual exposure to service-level penalties rather than around cost reduction in the abstract.

The recourse mechanism handles forecast-error variance for free in the operational sense, but it does not handle forecast-error bias for free: bias correction and variance reduction look like complements in an accuracy table but are substitutes in operational terms, and forecasting investment should be allocated primarily to bias reduction. The cost–emission trade-off revealed by the Pareto frontier is convex, with a corner-to-corner cost premium of approximately 22% for an emission reduction of approximately 18%, decomposing into a 9% green-routing premium and a 12% CVaR premium. The convexity identifies a green-leaning balanced configuration as the operationally attractive default, capturing most of the available decarbonisation benefit at a fraction of the corner-to-corner premium.

The framework’s three layers of uncertainty management—machine learning forecasting, two-stage recourse, and the CVaR constraint—perform operationally distinct jobs and are non-substitutable: recourse without CVaR keeps expected cost low but offers no tail-cost commitment, CVaR without recourse imposes a tail-cost bound on a deterministic plan that has no mechanism to comply with it, and better forecasting cannot substitute for either layer under realistic operating conditions. The three-layer separation is one of the central design properties of the framework and underwrites its operational coherence across the range of configurations tested in Chapter 5.

The implications for shipping lines, port operators, inland transport providers, and regulators are correspondingly distinct. Shipping lines benefit primarily from service-level reliability and from explicit visibility into the cost–emission trade-off; port operators benefit from inbound-volume visibility, modal-shift trend identification, and the alignment of robustness incentives with carriers; inland transport providers benefit from demand-side modal-mix signals and from the ceiling that the route-aware recourse penalty places on emergency-recourse premiums; regulators benefit from evidence that modal-shift incentives, emission-reporting focused on outcomes, and risk-aware planning adoption all support

the broader policy agenda of supply-chain decarbonisation and resilience.

The findings hold within the calibration boundary documented in Section 6.7.1, within the single-period horizon documented in Section 6.7.2, and under the cross-port independence simplification documented in Section 6.7.3. The three boundary conditions translate directly into the three future-work directions that the next chapter develops: validation on proprietary carrier data, multi-period rolling-horizon extension, and multivariate residual modelling that preserves cross-port covariance under joint shocks. The mechanistic findings of the present chapter are structural properties of the methodological design and would be expected to hold qualitatively across networks with similar multimodal structure; their quantitative magnitudes should be re-estimated on each new deployment context.

The chapter that follows revisits the research questions and hypotheses of Chapter 1 in the light of the present discussion, articulates the methodological and applied contributions of the thesis, and develops the three future-work directions in their own right.

Chapter 7

Conclusion and Future Work

This thesis has developed an integrated decision-support framework for empty-container repositioning across multi-port multimodal logistics networks, delivered as a working decision-support platform, and evaluated empirically on a calibrated five-port Italian network spanning 192 months of monthly data from January 2010 through December 2025. The framework couples a forecasting layer based on segment-specific machine learning with a two-stage stochastic optimisation layer that incorporates a Conditional-Value-at-Risk constraint and a route-aware recourse penalty, linked through an empirical scenario construction that grounds the optimisation in the realised forecast residuals rather than in parametric distributional assumptions. The framework is implemented as a working prototype with a ten-tab user interface and a dual-branch architecture that runs the back-testing and walk-forward pipelines in parallel.

This final chapter consolidates the empirical and methodological contributions of the thesis and identifies the directions in which the framework can be productively extended. Section 7.1 revisits the four research questions formulated in Chapter 1 in light of the empirical evidence developed across Chapters 5 and 6. Section 7.2 restates the methodological contributions, and Section 7.3 restates the applied contributions, in the form in which they have been validated by the case study. Section 7.4 summarises the practical implications for the principal stakeholder classes in the container-logistics ecosystem. Section 7.5 develops three future-work directions: validation on proprietary carrier data, a multi-period rolling-horizon extension, and a multivariate residual model that preserves cross-port covariance under joint shocks. Section 7.6 closes the thesis with a brief reflection on the broader operational and policy context in which the framework is positioned.

7.1 Revisiting the research questions

Chapter 1 organised the thesis around four research questions that targeted the predictive layer, the linkage between prediction and prescription, the prescriptive layer, and the engineering question of practical deployability. This section returns to each in turn and reports the empirical answer that the work has produced.

7.1.1 RQ1: forecasting heterogeneity

The question. RQ1 asked: *Which forecasting approach—among classical baselines, tree-based machine learning models, and a weighted ensemble—most accurately predicts empty container demand at the port–container-type level, and how stable is this ranking under operationally realistic out-of-sample evaluation?*

The empirical answer. The empirical evidence of Section 5.1 and the mechanistic interpretation of Section 6.1 together resolve RQ1 in two parts. At the family level, the four tree-based learners and the inverse-MAE ensemble dominate the three classical baselines uniformly: no classical baseline is ever selected by either the back-testing or the walk-forward protocol across the ten port–container segments of the case-study network, and pairwise Diebold–Mariano tests at the 3% significance level reject equal predictive accuracy in approximately 80% of the 280 within-segment comparisons. The family-level finding is statistically robust and generalises across the heterogeneous structural mix of surplus and deficit ports in the Italian Mediterranean network.

At the algorithm level, by contrast, the ranking is unstable. CatBoost emerges as the most frequent walk-forward winner, selected on six of the ten segments, with XGBoost selected on one segment and the inverse-MAE ensemble on the remaining three. Under back-testing the picture changes: LightGBM becomes the most frequent winner, with Random Forest selected on one segment and CatBoost on the other four. The back-testing and walk-forward protocols disagree on the best algorithm in eight of the ten segments, and the segments where they agree do so because CatBoost happens to win under both evaluation regimes rather than because the two protocols are interchangeable in general.

Hypothesis H1 confirmed. The finding confirms Hypothesis H1 of Section 1.4: no single forecasting method is uniformly superior across all port–container-type segments. The within-family ranking is unstable, the ranking depends on the evaluation protocol, and the operationally honest walk-forward protocol disagrees with back-testing on the majority of segments. The framework’s segment-specific selection rule responds to this heterogeneity by selecting the best-performing algorithm separately for each segment under each protocol, which is the operationally appropriate design choice given the empirical evidence.

7.1.2 RQ2: forecast–optimisation coupling

The question. RQ2 asked: *Does grounding stochastic scenario construction in the empirical residuals of a calibrated forecasting model translate into demonstrable operational benefits, and how does forecast accuracy propagate through the optimisation pipeline to influence operating cost and service-level reliability?*

The empirical answer. The empirical evidence of Section 5.2.3 and the mechanistic interpretation of Section 6.2 together resolve RQ2 with a finding that is both unambiguous and counterintuitive. The walk-forward protocol produces forecast errors that are, on average, roughly three times wider than the

back-testing errors on the same segments. A naive expectation would project this threefold deterioration in forecast accuracy onto a comparable deterioration in operating cost. The empirical evidence shows that the projection does not hold: expected cost under the walk-forward stochastic configuration (scenario S4) is approximately 0.6% higher than under the back-testing equivalent (scenario S2), while service-level reliability rises from approximately 97% to 99.5%.

The mechanism is the two-stage recourse channel. The optimiser does not respond to the wider residual distribution by inflating the first-stage commitment to insure against the wider range of realisations. It responds by reshaping the scenario-specific second-stage flows, absorbing the additional uncertainty through the recourse variables $y_j^{+, (s)}$ and $y_j^{-, (s)}$ rather than through the first-stage variable x_{ijm} . The forecast-error volume therefore manifests as recourse activity in the realised scenarios, not as a higher upfront expense in the planning commitment. The route-aware recourse penalty of Eq. (3.23) prices this absorption at the cheapest reachable emergency routing into each destination port, which keeps the recourse mechanism operationally coherent with the network economics rather than imposing an arbitrary shortage cost.

Hypothesis H2 confirmed. The finding confirms Hypothesis H2 of Section 1.4: demand scenarios constructed from empirical walk-forward residuals produce optimisation outcomes that are operationally aligned with the actual error characteristics of the deployed forecasting model, and the two-stage recourse channel absorbs forecast-error volume into reshaped second-stage flows so that improvements in forecasting quality are realised primarily as gains in service-level reliability rather than as proportional reductions in nominal first-stage cost. The recourse-as-absorber finding is one of the central operational contributions of the framework, and its testable consequence is that the deployment case for the framework should be built around contractual exposure to service-level penalties rather than around cost reduction in the abstract.

7.1.3 RQ3: cost–emission trade-offs

The question. RQ3 asked: *What are the achievable trade-offs between operational cost, service level, and CO₂ emissions when empty container repositioning is jointly optimised across road, rail, and short-sea transport modes, and how do these trade-offs respond to tail-risk control through a CVaR constraint?*

The empirical answer. The Pareto frontier of Section 5.3 and its mechanistic interpretation in Section 6.4 together resolve RQ3 in three parts. First, the corner-to-corner trade-off between the cost-optimal and emission-optimal configurations delivers an emission reduction of approximately 18% at a cost premium of approximately 22%, indicating that the multimodal network offers a substantial but non-trivial decarbonisation lever. Second, the 22% corner-to-corner premium decomposes into a green-routing premium of approximately 9%, isolated by the pairwise comparison between scenarios S5 and S6, and a CVaR premium of approximately 12%, isolated by the pairwise comparison between scenarios S4 and S6. The two premia are operationally distinct: the green-routing premium is the cost of substituting cleaner modes for dirtier ones at the routing decision, while the CVaR premium is the cost of insuring against the worst-5% tail of demand realisations. Third, the frontier is convex, with the marginal cost

of emission reduction lowest in the neighbourhood of the balanced configuration and highest near the emission-optimal corner; an operator targeting an emission reduction in the 10–15% range relative to the cost-optimal benchmark can achieve that target at a cost premium of approximately 5–10%, substantially below the 22% corner-to-corner premium.

The CVaR constraint, when active, acts as a distributional protection mechanism rather than as an expected-cost mechanism. The pairwise comparison between scenarios S4 (no CVaR) and S6 (with CVaR), both walk-forward and cost-focused, shows the expected cost rising by approximately 12% while the $\text{CVaR}_{0.95}$ tail measure contracts substantially. The robustness premium that the constraint charges buys distributional protection rather than expected savings, and its operational relevance grows sharply under high-impact disruptions such as a new pandemic wave, a geopolitical disruption that reroutes major trade lanes, or a sudden supply-side shock.

Hypothesis H3 confirmed. The finding confirms Hypothesis H3 of Section 1.4: multimodal optimisation across road, rail, and short-sea shipping enables the identification of Pareto-efficient operating points that simultaneously reduce cost and CO₂ emissions relative to single-mode baselines, and the cost–emission frontier is convex with meaningful emission reductions inexpensive in the neighbourhood of the balanced configuration but disproportionately expensive close to the green-optimal corner. The practical implication is that a green-leaning balanced configuration captures most of the available decarbonisation benefit at a fraction of the cost commitment of the corner itself, which is the operationally attractive default position along the frontier.

7.1.4 RQ4: decision-support platform feasibility

The question. RQ4 asked: *To what extent can the integrated forecast-driven stochastic optimisation pipeline be realised as a working, modular decision-support platform suitable for tactical planning use by port operators and shipping companies?*

The empirical answer. The platform documented in Section 3.9 of Chapter 3 and demonstrated through the ten-tab interface of Sections 3.2.6 operationalises the methodological framework as a working Python prototype on commodity hardware. The pipeline runs end-to-end—from raw-data ingest through forecasting, scenario construction, optimisation, and diagnostics—in approximately three and a half minutes on an Apple M3 workstation under the case-study configuration, as documented in the timing profile. The back-testing and walk-forward pipelines run in parallel through the dual-branch architecture, and the operationally honest figure is visible at every analytical step rather than being buried in a single summary statistic. The analytical centre of the platform exposes seven sub-views—scenarios, scenario charts, Monte Carlo robustness, forecast–optimisation coupling, CVaR sensitivity, scenario construction methods, and back-testing versus walk-forward comparison—that stress-test the framework along orthogonal methodological dimensions.

The platform is implemented entirely in open-source Python, with no proprietary dependencies that would obstruct deployment in operational environments. The linear-programming solver runs each optimisation in under half a second through the CBC backend, comfortably within the planning cadence of a

monthly operational cycle. The data-ingest tab accepts the same schema regardless of whether the input is the calibrated dataset of this thesis or a proprietary operational dataset, which means that the framework is structurally ready for deployment against carrier data once the necessary data-sharing arrangements are in place.

Hypothesis H4 confirmed. The finding confirms Hypothesis H4 of Section 1.4: the complete forecasting–optimisation pipeline has been implemented as a working decision-support platform with a modular software architecture that accepts standard operational inputs, runs end-to-end within operationally acceptable computation times on commodity hardware, and produces deployable repositioning plans. The binding constraint on live deployment is data-access, not framework feasibility: monthly port-level flows disaggregated by container type sit across carrier, terminal-operator, and customs systems that are rarely integrated, and most cross-system access is mediated by bilateral data-sharing agreements between specific parties (United Nations Conference on Trade and Development, 2023). The framework is ready for that access; the operational ecosystem is the layer that needs to catch up.

7.2 Methodological contributions

The thesis advances the methodology of empty-container repositioning along five dimensions that the case study has validated empirically.

7.2.1 Integrated forecast-driven stochastic optimisation framework

The first contribution is the integrated forecast-driven stochastic optimisation framework itself: a pipeline in which the forecasting and optimisation layers are coupled through empirical residual sampling rather than treated as independent sub-problems. The literature reviewed in Chapter 2 consistently treats the two layers in isolation, with forecasting studies stopping at predictive-accuracy metrics and optimisation studies treating demand uncertainty through parametric distributional assumptions (Abdelshafie et al., 2023; Sommer et al., 2025; Zugno et al., 2025). The present framework closes the gap by threading the empirical residual distribution of the selected forecasting algorithm directly into the scenario set that drives the optimisation, so that the predictive and prescriptive layers speak the same statistical language. The empirical evidence of Chapter 5—particularly the recourse-as-absorber finding—establishes that the coupling is not merely structurally elegant but operationally consequential: the layer integration produces outcomes that the parallel treatment of the two layers cannot.

7.2.2 Eight-algorithm cross-paradigm forecasting evaluation

The second contribution is the rigorous cross-paradigm evaluation of eight forecasting algorithms organised into three families—three classical baselines (Naive 3M, Drift, Seasonal Naive), four tree-based machine learning models (Random Forest, XGBoost, LightGBM, CatBoost), and one inverse-MAE weighted ensemble—applied to all ten port–container-type segments of the case-study network under

both back-testing and walk-forward protocols. Pairwise Diebold–Mariano testing across 280 within-segment comparisons confirms that the observed accuracy differences are statistically robust rather than sampling artefacts. To the author’s knowledge, this is the most comprehensive empirical comparison of forecasting paradigms reported in the empty-container repositioning context with full statistical-significance treatment, and it produces the within-family heterogeneity finding that underwrites the framework’s segment-specific selection rule.

7.2.3 The recourse channel as a forecast-error absorber

The third contribution is the explicit characterisation of the recourse channel as an absorber of forecast-error volume. The thesis demonstrates empirically that the two-stage stochastic structure absorbs forecast-error volume into reshaped second-stage flows scenario by scenario, so that improvements in forecasting accuracy translate primarily into gains in service-level reliability rather than into proportional reductions in nominal first-stage cost. The finding is articulated explicitly in Section 6.2 of Chapter 6, where it is identified as one of the central operational properties of the framework. The mechanism is rarely articulated explicitly in the empty-container literature (Vassos et al., 2025) and has direct operational consequences for the investment case in forecasting infrastructure under different contractual environments: the deployment case for the framework should be built around contractual exposure to service-level penalties rather than around cost reduction in the abstract.

The contribution extends to the bias–variance distinction developed in Section 6.3. The recourse-absorption mechanism works on the variance component of forecast error but not on its bias: an unbiased-but-spread forecast produces a balanced recourse profile that the framework absorbs as designed, while a biased forecast produces a one-sided recourse profile that the optimisation layer cannot correct. Bias correction and variance reduction look like complements in an accuracy table but are substitutes in operational terms, and forecasting investment under the framework should be allocated primarily to bias reduction. The distinction has been extensively developed in the machine-learning literature as a property of estimators (Hastie et al., 2009), but its operational consequences for stochastic logistics optimisation have received limited systematic attention, and the present thesis makes them explicit.

7.2.4 Route-aware recourse penalty formulation

The fourth contribution is the route-aware recourse penalty formulation of Eq. (3.23). The standard fixed-penalty convention in stochastic empty-container repositioning models applies a single penalty coefficient c^+ uniformly across all destination ports, irrespective of where the shortage occurs in the network. The thesis replaces this convention with a port-specific, network-consistent recourse cost that prices unmet demand at the cheapest reachable emergency repositioning into each destination, plus a service-delay surcharge that captures the operational disruption of off-cycle procurement. The construction has three operational properties that make it faithful to the economics of recourse: the optimiser cannot game the recourse cost by exploiting artificially low shortage penalties, the geographic distribution of unmet demand reflects the relative cost of covering shortages at each destination, and the comparison between deterministic and stochastic plans becomes methodologically clean because both are charged at the same

c_j^+ schedule. The formulation is a methodological refinement of the standard two-stage stochastic structure that preserves the linear-programming tractability of the original while removing a known weakness of the uniform-penalty convention.

7.2.5 Dual-branch BT/WF architecture as a methodological commitment

The fifth contribution is the dual-branch back-testing versus walk-forward architecture documented in Section 3.9.3, which runs both evaluation protocols in parallel and surfaces both estimates side by side at every analytical step. The architecture is not merely an interface convenience: it is a methodological commitment to auditability. The empirical evidence of Chapter 5 demonstrates that the two protocols disagree on the best forecasting algorithm in the majority of port–container segments and that the walk-forward forecast errors are approximately three times wider than the back-testing errors on the same segments. A platform that surfaced only one of the two protocols would either understate the realistic forecast error (if it surfaced back-testing alone) or discard the diagnostic value of the single-window view (if it surfaced walk-forward alone). The dual-branch architecture forces every analyst to confront the gap between the two protocols rather than allowing it to be silently absorbed into a single headline number, which is the operational signature of the framework’s commitment to honest deployment estimates.

7.3 Applied contributions

The thesis advances the applied side of the empty-container repositioning problem along three dimensions.

7.3.1 Working modular decision-support platform

The first applied contribution is the working modular decision-support platform itself, implemented in Python and organised as a ten-tab interface in which each pipeline stage—data ingest, forecasting, scenario generation, optimisation, results, and sensitivity diagnostics—is exposed as an independently auditable module. The platform runs end-to-end on commodity hardware under a fully open-source software stack, with no proprietary dependencies that would obstruct deployment in operational environments. The dual-branch architecture surfaces back-testing and walk-forward results in parallel, and the analytical centre exposes seven sub-views that stress-test the framework along orthogonal methodological dimensions: scenarios, scenario charts, Monte Carlo robustness, forecast–optimisation coupling, CVaR sensitivity, scenario construction methods, and back-testing versus walk-forward comparison. The platform is designed to support integration with existing port and carrier ERP environments rather than to replace them (Jahangard et al., 2025; Khazzar et al., 2025), and the data-ingest interface accepts the same schema regardless of whether the input is the calibrated dataset of this thesis or a proprietary operational dataset.

7.3.2 Empirical case study on a calibrated five-port Italian network

The second applied contribution is the empirical case study on the five-port Italian Mediterranean network (Genova, La Spezia, Livorno, Trieste, and Gioia Tauro), using 192 months of monthly data spanning three structurally distinct demand regimes: a pre-disruption baseline from 2010 to 2019, the COVID-19 shock and recovery from 2020 to 2022, and the post-pandemic period of geopolitical freight volatility from 2023 to 2025 (United Nations Conference on Trade and Development, 2023; Notteboom et al., 2021; Toygar et al., 2024). The case study quantifies the operational gains of the framework on a realistic logistics network: service level rises from 94.1% under the deterministic baseline to 99.5% under the walk-forward stochastic configuration, CO₂ emissions fall by approximately 18% under the green-focus configuration relative to the balanced reference at a routing premium of approximately 9%, and the CVaR constraint bounds the worst-5% tail of the cost distribution at a premium of approximately 12%. The results provide quantitative evidence that the framework can simultaneously improve service-level reliability, control tail-risk, and reduce CO₂ emissions on a network that is representative of the Italian and broader Mediterranean container-logistics ecosystem.

7.3.3 Three layers of uncertainty management

The third applied contribution is the operational characterisation of the framework's three layers of uncertainty management developed in Section 6.5. The framework treats demand uncertainty at three operationally distinct points: machine-learning forecasting reduces average forecast error and improves the central demand estimate, two-stage recourse absorbs the realised forecast error scenario by scenario, and the CVaR constraint bounds the right tail of the cost distribution. The thesis demonstrates that the three layers are non-substitutable: recourse without CVaR keeps expected cost low but offers no commitment about the worst-case scenario, CVaR without recourse imposes a tail-cost bound on a deterministic plan that has no mechanism to comply with it, and better forecasting cannot substitute for either layer under realistic operating conditions. The three-layer separation is one of the central design properties of the framework and underwrites its operational coherence across the range of configurations tested. The robustness premium that the CVaR constraint charges grows sharply under high-impact disruptions—a new pandemic wave, a geopolitical disruption that reroutes major trade lanes, or a sudden supply-side shock—which are precisely the regimes in which the framework's empirical advantage over the deterministic baseline is most pronounced.

7.4 Practical implications

The methodological and applied contributions of the thesis translate into a small set of practical implications that cut across the principal stakeholder classes in the container-logistics ecosystem. The detailed per-stakeholder analysis of Section 6.6 provides the full discussion; the present section synthesises the cross-stakeholder themes.

Service-level reliability over cost reduction. The framework's dominant operational return is service-level reliability rather than nominal cost reduction. The deployment case should therefore be built around contractual exposure to service-level penalties—liquidated-damages clauses on missed-delivery thresholds, service-level-agreement breaches with key customers—rather than around cost reduction in the abstract. The reliability gain documented in Section 6.2 converts directly into avoided penalties for operators with material contractual exposure, and into customer-relationship quality for operators with thinner exposure. The framework's value proposition differs across operators depending on the contractual environment in which they operate.

Bias-prioritised forecasting investment. Forecasting investment under the framework should be allocated primarily to bias reduction rather than to variance reduction. The recourse channel handles variance for free in the operational sense; it does not handle bias for free, and every unit of bias in the forecasting layer translates into a proportional unit of one-sided recourse activity that the optimisation layer cannot avoid. Operators investing in predictive analytics for repositioning should audit their forecasting layer for bias and prioritise bias-reducing model changes over generic accuracy improvements.

A green-leaning balanced configuration as the operational default. The convexity of the cost-emission Pareto frontier identifies a green-leaning balanced configuration as the operationally attractive default. The cheapest emission reductions are not at the green-optimal corner but in its neighbourhood, and an operator targeting an emission reduction of 10–15% relative to the cost-optimal benchmark can achieve that target at a cost premium of 5–10%, well below the 22% corner-to-corner premium. The framework makes the trade-off visible at the planning interface, which supports an explicit positioning decision along the frontier rather than a tacit one inherited from a single-objective optimiser.

Modal-shift signal for the inland transport ecosystem. The green-focused configuration produces a modal mix in which rail and short-sea shipping carry the bulk of the repositioning volume, with road absorbing the residual when the cleaner modes exhaust their capacity. The pattern is itself an investment signal for the inland transport sector: rail operators on the medium corridors and short-sea operators on the longer routes stand to gain volume from the green-leaning trend, while road hauliers stand to absorb the overflow. Ports with established intermodal infrastructure are positioned to benefit from the trend; ports without such infrastructure face a strategic gap as carriers adopt green-leaning planning frameworks.

Outcome-focused regulatory design. The framework's emission-reduction outcomes are governed by the objective weighting rather than by the optimisation paradigm or the forecasting protocol. The implication for regulatory design is that emission-reporting requirements should focus on the operational outcome (the realised modal mix, the realised emission footprint per TEU repositioned) rather than on the input configurations that produced it. The same framework operated under different objective weightings reports materially different emission outcomes; different frameworks operated under similar objective weightings report similar emission outcomes. Outcome-focused regulation aligns the incentive environ-

ment with the operational realities of multimodal repositioning and supports the broader decarbonisation policy agenda set out by the International Maritime Organization’s 2023 greenhouse-gas strategy (International Maritime Organization, 2023) and the European Union’s Emissions Trading System inclusion of maritime transport.

Risk-aware planning for supply-chain resilience. The CVaR-constrained configurations of the framework deliver the largest robustness contribution under high-impact disruptions, as Section 6.5.3 identified. Regulatory recognition that risk-aware planning frameworks deliver supply-chain resilience that expected-cost minimisation does not—through guidance, through public-sector contingency support conditioned on the use of such frameworks, or through procurement preferences for operators using risk-aware tools—would align private operational incentives with the public interest in supply-chain resilience. The post-pandemic period has demonstrated that container supply chains are exposed to high-impact disruptions that expected-cost minimisation alone cannot bound, and the framework’s empirical evidence supports the case for a regulatory environment that internalises this exposure into private planning incentives.

7.5 Future work

The boundary conditions identified in Section 6.7 translate into three future-work directions that the present framework is structurally ready to support. The directions are ordered by their consequence for the framework’s deployment readiness: validation on proprietary carrier data is the single most consequential extension, followed by the multi-period rolling-horizon formulation, followed by the multivariate residual model that preserves cross-port covariance under joint shocks.

7.5.1 Validation on proprietary carrier data

The framework has been developed and validated on the calibrated dataset documented in Section 3.2, anchored to publicly available aggregate statistics from Assoport and Eurostat rather than to raw carrier-level operational records. The calibration choice preserves the structural imbalances, seasonality, and disruption signatures of the underlying flows, and the methodological contributions of the thesis are robust to the data source. The relative comparisons across configurations—the 9% green-routing premium, the 12% CVaR premium, the 18% emission reduction under the green-focus configuration, the recourse-as-absorber finding—are structural properties of the framework and would be expected to hold qualitatively under proprietary data. The absolute EUR cost and emission figures, by contrast, are calibration-conditional and would change under proprietary parameterisation.

Validation on proprietary carrier data would deliver three forms of evidence that the calibrated case study cannot produce on its own. First, it would test whether the recourse-as-absorber mechanism survives under data whose residual profile reflects realised carrier operations rather than calibrated public statistics. The mechanism depends on the residuals being approximately unbiased, as Section 6.3 argued, and proprietary residual profiles may exhibit systematic bias that the calibrated dataset does not.

Second, validation on proprietary data would refine the absolute magnitudes of the framework’s operational outcomes for the specific carrier whose data feeds the validation: the service-level reliability, the cost–emission trade-off, and the CVaR tail measure would all acquire absolute interpretations grounded in the carrier’s own operational experience. Third, the validation would establish a deployment baseline against which subsequent operational outcomes under the framework could be measured, supporting a before-and-after evaluation of the framework’s operational impact.

The platform is structurally ready for this validation. The data-ingest tab of Section 3.2.6 accepts the same schema regardless of whether the input is the calibrated dataset or a proprietary operational dataset, and the subsequent pipeline depends on the schema rather than on the data source. The binding constraint is therefore not the framework itself but the data-sharing arrangements required to feed it: monthly port-level flows disaggregated by container type sit across carrier, terminal-operator, and customs systems that are rarely integrated, and most cross-system access is mediated by bilateral data-sharing agreements between specific parties. The natural deployment pathway is a research collaboration between the framework’s developers and a carrier or port-operator partner with the operational incentive to validate the framework against their own operational records, under a data-sharing agreement that protects commercial sensitivity. Such a collaboration would mark the transition of the framework from a research-grade prototype to an operationally validated planning tool.

7.5.2 Multi-period rolling-horizon extension

The framework is formulated as a single-period two-stage stochastic programme, and the opening stock of empty containers is treated as a known input at the start of the planning month. Inventory carry-over between consecutive planning months is not modelled, and the recourse channel is bounded by what can be absorbed within a single month. A natural extension is a multi-period rolling-horizon formulation in which the framework optimises across a planning horizon of several months, with the decisions of each month feeding the opening stock of the next and the recourse mechanism extending across months rather than being confined to within a single month.

The mathematical structure of the extension is the standard rolling-horizon stochastic programme: the planning horizon is partitioned into T consecutive months, each with its own demand scenarios drawn from the forecasting residual pool, and the first-stage decisions of month t become the boundary conditions for month $t + 1$. The optimisation horizon rolls forward as time advances: at the start of each month the planner re-solves the multi-period problem over the next T months, commits the first-month decisions to execution, and discards the optimisation for months 2 through T when the next instance is solved. The single-period structure of the present framework is the special case $T = 1$.

The extension would change the framework’s operational outcomes in two predictable directions. First, the operational value of the stochastic optimisation relative to the deterministic baseline would likely grow, because the recourse channel would acquire an additional degree of freedom: a shortage at a destination port in one month could be partly mitigated by adjusting the inbound flow in the following month, which the single-period formulation cannot accommodate. Second, the treatment of residual auto-correlation in the scenario construction would become operationally consequential: the block-bootstrap

variant of Section 5.5.2 which is near-equivalent to the independent-sampling baseline at the single-period horizon, would likely become operationally distinguishable under the multi-period formulation because residual autocorrelation would propagate through successive planning months. The methodological choice between independent sampling and block bootstrap, currently operationally inconsequential, would acquire operational stakes in the multi-period setting.

The empirical magnitude of these changes is an open question that the multi-period extension would be designed to resolve. The framework’s operational outcomes under a multi-period formulation may improve substantially or only marginally relative to the single-period baseline, depending on the strength of the cross-month dependence in the realised demand series and on the cost structure of the recourse arcs that connect months. Proprietary multi-period data would be required to estimate these magnitudes with confidence, which links the multi-period extension naturally to the proprietary-validation direction discussed in Section 7.5.1.

7.5.3 Multivariate residual modelling under joint shocks

The empirical scenarios of the present framework are constructed by sampling residuals independently across ports. The independence assumption is a substantive simplification that under-represents the cross-port co-movement that arises under system-wide shocks: during a global disruption such as the COVID-19 wave, forecast errors tend to be positively correlated across the network, and independent draws across ports therefore under-estimate the tail-risk that the CVaR constraint is designed to bound. The $\text{CVaR}_{0.95}$ measures reported in Table 5.6 are consequently conservative-on-the-low-side estimates of the true tail-risk under realistic correlated demand shocks, and the recourse-as-absorber mechanism is similarly optimistic under cross-port co-movement.

A multivariate residual model that preserves the empirical cross-port covariance is the natural extension that would address this boundary. The construction would replace the per-segment residual pool of Eq. (3.16) with a joint residual sample that captures the contemporaneous correlation across ports, and the scenario construction of Eq. (3.17) would draw from the joint sample rather than from independent per-segment samples. Three implementation options present themselves. The simplest is a parametric multivariate Gaussian fit to the joint residual distribution, which would preserve the contemporaneous covariance at the cost of smoothing away the heavy-tailed structure that Section 5.5.1 identified as operationally important. A second option is a copula-based construction, in which the marginal residual distributions are preserved empirically while the cross-port dependence structure is captured through a fitted copula (Gaussian, t, or vine); this preserves both the marginal heavy tails and the contemporaneous dependence, at the cost of additional modelling complexity. A third option is direct empirical sampling from contemporaneous residual vectors, which preserves both the marginal distributions and the joint dependence non-parametrically but at the cost of reduced sample size, since each calendar month contributes only one joint residual vector.

The methodological commitment of the framework to empirical scenario construction (Section 3.4.2) suggests that the third option—direct empirical sampling from joint residual vectors—is the natural extension of the present approach, with the parametric and copula-based alternatives as benchmarks against

which the empirical method can be evaluated. The tail-risk estimates produced under this construction would tighten relative to the independent baseline, and the CVaR constraint would deliver more accurate protection against system-wide shocks. The empirical magnitude of the tightening is an open question; it would be expected to grow with the strength of the underlying cross-port residual correlation and to be largest under the disruption regimes that the present framework cannot fully represent.

7.6 Closing remarks

The empty-container repositioning problem sits at the intersection of three operational pressures that have intensified over the past five years and that show no sign of easing. The first is the persistent structural imbalance between import and export trade lanes that has produced the twenty-plus-percent empty-flow share of global container movements that the literature has documented since the 1990s and that the present case study confirms on the Italian Mediterranean network. The second is the supply-chain volatility that the COVID-19 pandemic, the subsequent inflationary period, and the geopolitical disruptions of the early 2020s have introduced into demand patterns that were previously analysable as quasi-stationary cycles around predictable seasonal envelopes (Notteboom et al., 2021; United Nations Conference on Trade and Development, 2023; Toygar et al., 2024). The third is the emerging decarbonisation pressure on maritime and land transport that the International Maritime Organization’s 2023 greenhouse-gas strategy and the European Union’s Emissions Trading System inclusion of maritime transport have made into a binding operational constraint rather than a voluntary aspiration. The three pressures act jointly: the structural imbalance generates the flow volume that must be repositioned, the volatility makes the repositioning decision harder to plan in advance, and the decarbonisation constraint narrows the operational envelope within which the repositioning can take place.

The integrated framework developed in this thesis is a response to the joint operational pressure. The machine-learning forecasting layer extracts the residual predictability that remains in the disrupted demand series; the two-stage stochastic optimisation layer absorbs the residual unpredictability into re-shaped second-stage flows rather than into inflated first-stage commitments; the CVaR constraint bounds the tail-cost exposure that the post-pandemic disruption regime has made operationally significant; and the multimodal optimisation across road, rail, and short-sea shipping exposes the cost–emission trade-off that the decarbonisation pressure has made into a binding operational consideration. The framework is not a final answer to the empty-container repositioning problem, and the boundary conditions identified in Section 6.7 mark the limits of the empirical evidence it produces. But the methodological commitments that organise the framework—honest evaluation through walk-forward validation, route-aware recourse penalties, the empirical residual sampling, the three-layer separation of uncertainty management—offer a coherent operational response to the operational pressure under which the container-logistics ecosystem now operates.

The deployment readiness of the framework is constrained by data access rather than by methodological or computational limitations. The pipeline runs in under five minutes on commodity hardware, the linear-programming solver completes each optimisation in under half a second, the software stack is fully open-source, and the data-ingest interface accepts the same schema regardless of whether the

input is calibrated or proprietary. The framework is, in this technical sense, ready for deployment. The operational ecosystem in which the framework would be deployed is the layer that needs to catch up, through the data-sharing arrangements that link carrier, terminal-operator, and customs systems into a coherent operational picture and that the present fragmentation of those systems makes difficult to assemble. The methodological contributions of this thesis, alongside the working decision-support platform that implements them, are the author's response to this operational and policy challenge.

Bibliography

- Abdelshafie, A., Salah, M., Kramberger, T., and Dragan, D. (2023). Repositioning and optimal re-allocation of empty containers: A review of methods, models, and applications. *Sustainability*, 15(9):7023.
- Assoporti (2023). Movimenti portuali 2010–2023: Annual port statistics.
- Benghalia, A., Ferdjallah, A., Oudani, M., and Boukachour, J. (2023). Machine learning and simulation for efficiency and sustainability in container terminals. *Sustainability*, 15(3):2446.
- Braekers, K., Janssens, G. K., and Caris, A. (2011). Challenges in managing empty container movements at multiple planning levels. *Transport Reviews*, 31(6):681–708.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.
- Cai, M., Li, H., Guo, Z., and Lu, S. (2023). Data-driven empty container repositioning for large-scale railway networks with fuzzy demands. *IEEE Transactions on Fuzzy Systems*, 31(2):557–570.
- Cao, Z., Qian, T., Zhang, S., Song, H., and Tian, Y. (2023). Multi-port liner ship routing and scheduling optimisation using machine learning forecast and branch-and-price algorithm. *Transportation Research Part E: Logistics and Transportation Review*.
- CE Delft (2020). STREAM freight transport 2020: Emissions of freight transport modes. Technical report, CE Delft, Delft, The Netherlands.
- Chan, H. K., Xu, S., and Qi, X. (2019). A comparison of time series methods for forecasting container throughput. *International Journal of Logistics Research and Applications*, 22(3):294–303.
- Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.
- Diaz, R., Talley, W., and Tulpule, M. (2011). Forecasting empty container volumes. *The Asian Journal of Shipping and Logistics*, 27(2):217–236.
- Diebold, F. X. and Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3):253–263.

- Dragan, D., Keshavarzsaleh, A., Intihar, M., Popović, V., and Kramberger, T. (2020). Throughput forecasting of different types of cargo in the Adriatic seaport Koper. *Maritime Policy & Management*, 47(7):943–958.
- Epstein, R., Neely, A., Weintraub, A., Valenzuela, F., Hurtado, S., Gonzalez, G., Beiza, A., Naveas, M., Infante, F., Alarcon, F., Angulo, G., Berner, C., Catalan, J., Gonzalez, C., and Yung, D. (2012). A strategic empty container logistics optimization in a major shipping company. *Interfaces*, 42(1):5–16.
- Farhan, J. and Ong, G. P. (2018). Forecasting seasonal container throughput at international ports using SARIMA models. *Maritime Economics & Logistics*, 20(1):131–148.
- Gouaboua, A. C. F., Al-Kharaza, M., Hakimi, F., Khaled, T., and Amzil, K. (2024). Forecasting empty container availability for vehicle booking system application. *Procedia Computer Science*, 246:3103–3112.
- Harvey, D., Leybourne, S., and Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, 13(2):281–291.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2nd edition.
- Henesey, L. and Liu, Y. (2020). Forecasting in repositioning of empty containers — Analysis of machine learning approaches. In *Proceedings of the IAME Annual Conference*, Hong Kong.
- Holguín-Veras, J., Kalahasthi, L., Campbell, S., González-Calderón, C. A., and Wang, X. C. (2021). Freight mode choice: Results from a nationwide qualitative and quantitative research effort. *Transportation Research Part A: Policy and Practice*, 143:78–120.
- Hyndman, R. J. and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688.
- IFEU and INFRAS and IVE (2024). EcoTransIT World: Environmental methodology and data update 2024. Technical report, EcoTransIT World Initiative, Heidelberg, Germany.
- International Maritime Organization (2023). 2023 IMO Strategy on Reduction of GHG Emissions from Ships. Resolution MEPC.377(80) MEPC 80/WP.12, Annex 1, International Maritime Organization, London, United Kingdom. Adopted on 7 July 2023 by the Marine Environment Protection Committee.
- Jahangard, M., Xie, Y., and Feng, Y. (2025). Leveraging machine learning and optimization models for enhanced seaport efficiency. *Maritime Economics & Logistics*, 27:710–751.
- Janić, M. (2007). Modelling the full costs of an intermodal and road freight transport network. *Transportation Research Part D: Transport and Environment*, 12(1):33–44.
- Karmelić, J., Dundović, □., and Kolanović, I. (2012). Empty container logistics. *PROMET – Traffic & Transportation*, 24(3):223–230.

- Ke, G., Meng, Q., Finley, T., et al. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30.
- Khazzar, A., Sekaki, Y., Lachhab, Y., and El-Marzouki, S. (2025). Artificial intelligence in port logistics: A bibliometric analysis of technological integration and research dynamics. *International Journal of Accounting, Finance, Auditing, Management and Economics*, 6(10):662–679.
- Kowshik, R. R. (2023). Optimizing empty container repositioning at inland terminals: Using data analytics to make an informed decision for a seaport in Sweden. Technical Report E2023-017, Chalmers University of Technology.
- Kulshrestha, A., Yadav, A., Sharma, H., and Suman, S. (2024). A deep learning-based multivariate decomposition and ensemble framework for container throughput forecasting. *Journal of Forecasting*.
- Lam, J. S. L. (2014). Empty container logistics optimization: An implementation framework and methods. *Maritime Policy & Management*, 42(1):1–16.
- Lee, G.-C. and Bang, J.-Y. (2024). Forecasting container throughput of Singapore Port considering various exogenous variables based on SARIMAX models. *Forecasting*, 6:748–760.
- Leng, S., Wang, Y., Zhou, Y., Feng, X., Lee, P. T. W., and Bai, R. (2025). The growing role of artificial intelligence in smart container ports: Its application and future research directions. *Journal of International Logistics and Trade*. in press.
- Martius, C., Kretschmann, L., Zacharias, M., Jahn, C., and John, O. (2022). Forecasting worldwide empty container availability with machine learning techniques. *Journal of Shipping and Trade*, 7(1):19.
- Ministero delle Infrastrutture e dei Trasporti (2024). Valori indicativi di riferimento dei costi di esercizio dell'impresa di autotrasporto per conto di terzi. Technical report, Ministero delle Infrastrutture e dei Trasporti, Rome, Italy.
- Munim, Z. H., Fiskin, C. S., Nepal, B., and Chowdhury, M. M. H. (2023). Forecasting container throughput of major Asian ports using the Prophet and hybrid time series models. *The Asian Journal of Shipping and Logistics*.
- Newey, W. K. and West, K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55(3):703–708.
- Notteboom, T., Pallis, A., and Rodrigue, J.-P. (2021). Disruptions and resilience in global container shipping and ports: The COVID-19 pandemic versus the 2008–2009 financial crisis. *Maritime Policy & Management*, 48(4):421–448.
- Octaviansyah, D., Buchari, E., Arliansyah, J., and Nawawi, N. (2024). Multi-criteria analysis as a method for selecting the best route of hinterland connections: Case study in South Sumatra, Indonesia. *The Asian Journal of Shipping and Logistics*, 40:22–29.

- Olivo, A., Zuddas, P., Di Francesco, M., and Manca, A. (2005). An operational model for empty container management. *Maritime Economics & Logistics*, 7(3):199–222.
- Prokhorenkova, L., Gusev, G., Vorobev, A., et al. (2018). CatBoost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, volume 31.
- Ratmoko, H. and Zuliarso, E. (2025). Time series forecasting for container throughput using SARIMA and LSTM: A case study of Tanjung Emas Port, Semarang. *Maritime Technology and Society*, 4(3):232–242.
- Rockafellar, R. T. and Uryasev, S. (2000). Optimization of conditional value-at-risk. *Journal of Risk*, 2(3):21–41.
- Sadaghi, S. M., Shivaifar, I., and Rastad, M. (2024). Optimization of empty container operations by demand prediction and simulation: Case study of Shahid Rajaei Port. *International Journal of Coastal, Offshore and Environmental Engineering*, 1(2):38–52.
- Saeed, N., Nguyen, S., Cullinane, K., Gekara, V., and Chhetri, P. (2023). Forecasting container freight rates using the Prophet forecasting method. *Transport Policy*.
- Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, 2:160.
- Sharma, A. K. S. and Gupta, A. K. (2017). Importance of freight mode choice criteria: An MCDA approach. *Transportation Research Procedia*, 25:4619–4633.
- Shintani, K., Imai, A., Nishimura, E., and Papadimitriou, S. (2007). The container shipping network design problem with empty container repositioning. *Transportation Research Part E: Logistics and Transportation Review*, 43(1):39–59.
- Smart Freight Centre (2023). Global Logistics Emissions Council framework for logistics emissions accounting and reporting, version 3.0. Technical report, Smart Freight Centre, Amsterdam, The Netherlands.
- Sommer, B., Lee, S., Köhler Holst, K., and Boomsma, T. K. (2025). Markov decision processes for inland empty container inventory management. *Computational Management Science*, 22:24.
- Song, D.-P. and Carter, J. (2009). Empty container repositioning in liner shipping. *Maritime Policy & Management*, 36(4):291–307.
- Song, D.-P. and Dong, J.-X. (2012). Empty container repositioning. *Maritime Policy & Management*, 39(4):347–369.
- Song, J., Tang, X., Wang, C., Xu, C., and Wei, J. (2022). Optimization of multi-port empty container repositioning under uncertain environments. *Sustainability*, 14(21):13255.

- Stahlbock, R. and Voß, S. (2010). Improving empty container logistics: Can it avoid a collapse in container transportation? In *Operations Research Proceedings 2009*, pages 121–126, Berlin. Springer.
- Taşyürek, E. Y., Yılmaz, İ., Topgül, M., Solmaz, C., Omurca, S. İ., and Çakıl, L. (2025). LSTM-based forecasting workflow for container supply and demand in maritime logistics operations. In *Proceedings of the 6th International Conference on Computers and Artificial Intelligence Technology (CAIT 2025)*, pages 454–459.
- Theofanis, S. and Boile, M. (2009). Empty marine container logistics: Facts, issues and management strategies. *GeoJournal*, 74(1):51–65.
- Toygar, A., Yildirim, U., and Nart, S. (2024). Understanding the impact of global challenges on container shipping: A qualitative study in the COVID-19 era. *WMU Journal of Maritime Affairs*, 23:257–282.
- United Nations Conference on Trade and Development (2023). Review of maritime transport 2023. Annual report, United Nations Conference on Trade and Development, Geneva.
- Vassos, G., Lusby, R., and Pinson, P. (2025). Integrated optimization of operations and capacity planning under uncertainty for drayage procurement in container logistics. Preprint, submitted to Omega.
- Xu, S., Zou, S., Huang, J., Yang, W., and Zeng, F. (2022). Comparison of different machine learning approaches with conventional methods on container throughput forecasting. *Applied Sciences*, 12(19):9730.
- Zain, R. M., Rahman, M. N. A., Nopiah, Z. M., and Saibani, N. (2014). Understanding of empty container movement: A study on a bottleneck at an off-dock depot. In *AIP Conference Proceedings*, volume 1613, pages 403–419. American Institute of Physics.
- Zheng, S., Sha, J., Kong, Y., and Wang, Y. (2025). Research on artificial intelligence-driven container relocation problem for green ports. *Frontiers in Marine Science*.
- Zhou, C., Ma, N., Cao, X., Lee, L. H., and Chew, E. P. (2021). Classification and literature review on the integration of simulation and optimization in maritime logistics studies. *IIE Transactions*, 53(10):1157–1176.
- Zugno, M., Petersen, L., Ponikiewski, W., Szulc-Wodarska, E., and Tastu, J. (2025). Forecasting for empty container repositioning. *Maritime Economics & Logistics*, 23(1):1–27.