

Università degli Studi di Genova
Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi

**Design of methods and tools for the optimal planning of
maritime terminal processes**

by

Rexhina Hoxha

Theses Series

DIBRIS-TH-2026-XXXVIII

DIBRIS, Università di Genova

Via Opera Pia, 13 16145 Genova, Italy

<http://www.dibris.unige.it/>

Università degli Studi di Genova

Dipartimento di Informatica, Bioingegneria,

Robotica ed Ingegneria dei Sistemi

**Ph.D. Thesis in Computer Science and Systems Engineering
Systems Engineering Curriculum**

**Design of methods and tools for the optimal
planning of maritime terminal processes**

by

Rexhina Hoxha

July, 2026

**Department of Informatics, Bio-engineering, Robotics and Systems Engineering
University of Genoa**

DIBRIS, Univ. di Genova
Via Opera Pia, 13
I-16145 Genova, Italy
<http://www.dibris.unige.it/>

**Ph.D. Thesis in Computer Science and Systems Engineering
Systems Engineering Curriculum
(S.S.D. ING INF/04)**

Submitted by Rexhina Hoxha
Date of submission: 08/07/2026

Title: Design of methods and tools for the optimal planning of maritime terminal processes

Advisors: Prof. Simona Sacone¹ Dr. Enrico Zero¹ Ing. Stefano Cangelosi²

¹Department of Informatics, Bio-engineering, Robotics and Systems Engineering,
University of Genoa, Italy

² Department of Business Process Management, PSA Italy, Italy

Ext. Reviewers:

Prof. Claudia Caballini ³

Prof. Dritan Nace ⁴

³ Department of Environment, Land and Infrastructure Engineering,
Polytechnic University of Turin, Italy

⁴ Heudiasyc, University of Technology of Compiègne, France

Abstract

This thesis focuses on the optimization of selected operational processes within a port container terminal. The work has been developed in close collaboration with PSA Genova Pra' terminal, one of the major container terminals in Northern Italy. The terminal is modeled as a system of systems, comprising the berth, yard, gate, and rail yard subsystems, which interact to achieve the common objective of efficiently handling containerized cargo flows. Within this framework, the thesis concentrates on the gate and rail subsystems, which represent the primary interfaces between the terminal and the inland transport network. The main focus is to optimize processes associated with these subsystems through the application of prediction, optimization, and simulation methodologies, while accounting for their interactions with other terminal components, such as the yard and berth. For the gate subsystem, the thesis proposes predictive models for container arrivals and departures by truck. The predicted demand is then analyzed to evaluate its impact on terminal congestion and emissions. Based on this analysis, optimization strategies are developed to proactively redistribute truck arrivals, with the aim of mitigating congestion in yard areas and reducing emission peaks on critical days. The rail subsystem is addressed from two perspectives: the optimization of train discharge operations and the assessment of overall rail yard capacity. Although initial analytical and optimization models are developed, the inherent complexity of the system necessitates simplifying assumptions. To overcome these limitations, detailed simulation models are implemented, allowing a more realistic representation of operational constraints and interactions. Simulation results demonstrate improved accuracy in capacity evaluation compared to purely mathematical approaches. Overall, the methodologies and results presented in this thesis provide a structured decision-support framework for port terminal operators, illustrating how integrated prediction, optimization, and simulation tools can effectively support operational planning. Furthermore, this work highlights the value of close collaboration between academic research and industry in addressing real-world operational challenges in port terminals.

Acknowledgments

I am fortunate to have many people to thank, individuals who have shaped this journey and made my PhD experience one of the most meaningful and, I would even say, one of the most beautiful chapters of my academic path.

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Simona Sacone. Thank you for everything you have taught me, for your constant guidance, your invaluable advice, and for always finding the time to support me. I am especially grateful for your patience in answering my countless questions, always with kindness, calmness, and a smile. Being your PhD student has been a privilege and an honour. Beyond being an outstanding professor, you are an extraordinary person and an inspiring woman whom many would aspire to be like. And no matter how many more times you ask me to call you Simona, you will always be my Prof.

Although not officially my supervisor, I could not miss the opportunity to express my deepest gratitude to Prof. Roberto Sacile. It is difficult to put into words how much your support has meant to me throughout these years. You are not only a remarkable professor, but above all, one of the most wonderful people I have ever had the privilege of meeting. Even without always realizing it, you have always been there with your advice, your support, and your willingness to help, constantly creating opportunities that have shaped both my professional and personal growth. Thank you for believing in me and for always encouraging me to aim higher. I do not know if I will ever be able to fully repay everything you have done for me, Prof, but I will always be grateful.

My sincere thanks also go to my supervisor, Prof. Enrico Zero, who has been one of my greatest sources of support throughout these PhD years. Thank you for always being present, available, and ready to help whenever I needed it. I am especially grateful for your patience, your encouragement, and for guiding me through the most challenging moments, always helping me regain confidence when I needed it most.

Your guidance and constant presence have meant a great deal to me throughout this journey, and I am truly grateful for everything you have done for me. And thank you for never giving up on reminding me of every deadline, even when I was not always the best at respecting them. Looking back, I know every reminder was simply another way of looking out for me.

Last but certainly not least, a very important part of this journey has been the time I spent at PSA Genova Pra'. I would like to express my sincere gratitude to my industrial supervisor, Eng. Stefano Cangelosi. Thank you for everything you have patiently and generously taught me about the world of port terminals, especially considering that I was still new to this field. I am truly grateful for the many research ideas inspired by your insight, and for your constant support and positivity throughout this entire journey. Thank you for always believing in me and for placing your trust in me. I truly hope to live up to that trust as I continue my professional journey at PSA Genova Pra', a journey that began under your supervision and that I am proud to continue today.

One of the most special and rewarding experiences of my PhD was the five-month research stay I spent at PSA Antwerp, where I had the privilege of working under the supervision of Dr. Geert Van de Weyer. Thank you for everything you taught me, for your guidance, and for your constant support and encouragement throughout this experience. I will always be grateful for the opportunities you gave me.

I would also like to sincerely thank Roxan De Ridder, who, with great patience, guided me throughout my stay, answered every question I had, and helped me every step of the way. Beyond that, I am grateful for your positivity and kindness. I found in you not only a wonderful colleague, but also a friend, or, as I like to say, my favourite local in Antwerp.

To both of you, thank you for introducing me to the exciting world of FlexTerm and simulation, and for making my time at PSA Antwerp such a special and rewarding part of my PhD.

A special thank you goes to all my lab mates, people I would never simply call colleagues, but friends. We shared not only long hours in the lab, but also countless unforgettable experiences outside of it. You are among the most special people I have met, and I feel incredibly lucky and grateful to have crossed paths with each and every one of you.

Elisa, or as we call you, "my twin", thank you for every single moment we have shared together, in the lab, at conferences, and beyond. Most importantly, thank you for your friendship and for always listening to me whenever I needed you. Ti voglio un mondo di bene.

Thank you, Alessandro and Simone, for being so special and amazing. From the very first day I joined the lab, you made me feel part of it, never like an outsider or

someone from another country, but as someone who truly belonged. That is something I will never forget.

Thank you, Tommy, Yassine, Dani, and my sweet Hanie, for your friendship, for all the beautiful moments, and for all the adventures we went through together as immigrants. You are all incredibly special to me, and sharing these PhD years with you has made this journey so much more meaningful.

Thank you, Lorenzo, for your valuable friendship and for every fun activity, trip, and hiking adventure you talked us into. Thank you, Matteo, for your friendship and for our long, but always valuable, conversations. Thank you as well to Luca, Giulio, Giacomo, and Mohammad for all the good times in the lab.

And thank you, Abdellatif and Said, for all the laughter and the wonderful memories we created together during conferences, both in my country and in yours.

A special thank you goes to the professors and colleagues with whom I had the pleasure of collaborating and sharing so many memorable experiences during conferences and other academic events, especially Professors Silvia, Cecilia, and Massimo.

I would like to sincerely thank Professors Elinda and Ilir, who have always been close to me throughout this journey, supporting me with their kindness, encouragement, and positive energy.

I am also grateful to all the professors and colleagues I had the opportunity to meet at conferences around the world. Some of those encounters grew into friendships that I truly value, and I feel incredibly fortunate to have met such inspiring people along the way.

Moving to the second environment where I spent a significant part of my PhD, I would like to thank all my colleagues at PSA Genova Pra', who, over the years, guided me and helped me become part of the port world.

In particular, I would like to thank my colleagues from the BPM office, especially Ilaria for all her guidance, kindness, and support; Daniele for everything he taught me and for always being available to share his knowledge and answer my questions; and Andrea for always being so kind and helpful whenever I needed it. I would also like to thank my colleagues from the Operations and IT departments, including Paolo, Stefano Belcredi, Renzo, Riccardo, Anselmo, Alessandro, Fabio and many others with whom I had the pleasure of collaborating. Thank you also to Francesca and Giulia for your friendship and for all the nice moments we shared. I am grateful not only for your support and collaboration, but also for the kindness and warm welcome you have always shown me.

A very, very special thank you goes to my closest friend, Kristina, who has been by my side since the very first day of university and is still here, nine years later. You are the person I have turned to every single time I needed someone, knowing

that you would always be there for me, no matter what. I feel incredibly lucky to have someone as special as you in my life, with whom I have shared not only every challenge, but also every joyful moment. In every achievement of mine, no matter how small, there is a part of you.

A heartfelt thank you also goes to my roommates, Ledianë, Ina, and Anjeza, with whom I shared a home throughout these years. Thank you for all the beautiful moments we shared and for being such an incredible source of support. I am truly grateful for your understanding, your kindness, and for making our home feel like a family.

Thank you also to Ronaldo, Endrit, and Enton for your friendship and for always being ready to help whenever I needed it. They say that true friends reveal themselves in difficult times, and you have certainly proven that to me.

Një falënderim shumë i veçantë dhe i përzemërt shkon për familjen time të madhe. Faleminderit kushërinjve të mi, të cilët më kanë qëndruar pranë dhe kanë qenë gjithmonë aty për mua, duke më dhënë një mbështetje të jashtëzakonshme, veçanërisht në fillimet e këtij rrugëtimi në një vend krejtësisht të ri. Tek ju kam gjetur gjithmonë ngrohtësinë e shtëpisë. Për këtë do të jem përjetësisht mirënjohëse ndaj dada Klodit dhe familjes së saj, Majlindës dhe familjes së saj, Emiljanos dhe Gertit. Faleminderit edhe Anisës, personit tek i cili jam drejtuar sa herë kam pasur një vendim të vështirë për të marrë.

Dhe së fundmi, faleminderit motrave të mia, Xhemiles dhe Samantës, pa të cilat do të isha e humbur në këtë botë. Faleminderit prindërve të mi të mrekullueshëm për çdo sakrificë që keni bërë për mua, për çdo mbështetje, për çdo fjalë inkurajimi dhe për dashurinë tuaj të pakushtëzuar. Çdo sukses që arrij dhe çdo diplomë që marr ju dedikohet juve. Shpresoj me gjithë zemër t'ju kem bërë krenarë, sepse gjithçka që jam sot është falë jush. Thjesht, ju dua!

Table of Contents

List of Figures	11
List of Tables	16
Introduction	18
Chapter 1 Introduction to Port Terminals	27
1.1 The terminal as a system-of-systems	28
1.1.1 Berth operations	28
1.1.2 Gate operations	29
1.1.3 Rail operations	30
1.1.4 Yard operations	30
1.2 PSA International	31
1.3 PSA Genova Pra' terminal	33
1.3.1 History	33
1.3.2 Location and strategic role	34
1.3.3 Infrastructure and capacity	35
1.3.4 Main container areas	36
1.3.5 Handling equipment	37
1.3.6 Key challenges in PSA GP terminal operations	39
Chapter 2 Literature Review	41

2.1	Truck subsystem in container terminals	41
2.2	Rail subsystem in container terminals	47
Chapter 3 System Architecture		52
3.1	Subsystems and their interactions at PSA Genova Pra'	52
3.1.1	Export process via truck in gate subsystem	54
3.1.2	Import process via truck in gate subsystem	58
3.1.3	Train entry and exit in rail subsystem	59
3.1.4	Train discharge in rail subsystem	60
Chapter 4 Methodology		63
4.1	Predictive models	63
4.1.1	Cross-validation strategy	66
4.2	Emission estimation models	67
4.2.1	COPERT model	69
4.3	Optimization models	71
4.3.1	Mathematical programming	72
4.4	Scheduling framework	74
4.4.1	Scheduling environment classification	76
4.4.2	Resolution approaches	76
4.4.2.1	Exact approach	77
4.4.2.2	Heuristics and metaheuristics	77
4.5	Queuing-based system representation	80
Chapter 5 Simulation		82
5.1	Simulation for decision making	84
5.2	Discrete-event simulation	86
5.3	Simulation software environment	87
5.3.1	FlexSim	87

5.3.2	FlexSim environment terminology	88
5.3.3	Building model logic	90
5.3.4	Modeling randomness	92
5.3.5	Simulation experiments and result analysis	92
5.3.6	FlexTerm	93
5.3.6.1	FlexTerm modeling paradigm	93
5.3.6.2	Modeling rail yard in FlexTerm	95
5.3.6.3	Travel Manager	96
Chapter 6	Container Volume Prediction	97
6.1	Prediction of export containers via truck	99
6.1.1	Brief reminder of the export process	100
6.1.2	Artificial neural network predictive model	102
6.1.2.1	Leave-one-out cross-validation	105
6.1.2.2	Post-processing procedure	105
6.1.3	Heuristic predictive model	106
6.1.4	Export data description	107
6.1.5	Export prediction results	109
6.2	Prediction of import containers via truck	111
6.2.1	Brief reminder of the import process	111
6.2.2	Artificial neural network predictive model	112
6.2.2.1	Post-processing procedure	114
6.2.3	Import data description	115
6.2.4	Import prediction results	117
Chapter 7	Truck Operations Optimization	120
7.1	Optimization of truck arrival patterns to reduce congestion	120
7.1.1	First phase of the optimization scheme: redistribution in days	121

7.1.2	Second phase of the optimization scheme: redistribution along hours of the day	124
7.1.3	Results of the optimization schemes	125
7.2	Implementation of emission control strategies	130
7.2.1	Case study	130
7.2.2	Factors for emission factor estimation	132
7.2.3	Redistribution algorithm	133
7.2.4	Emission optimization problem	138
Chapter 8	Rail Operation Optimization	142
8.1	Optimized scheduling of train discharge	143
8.1.1	Current way of operating	143
8.1.2	Exact solution	145
8.1.2.1	Flexible flow-shop scheduling problem definiton	145
8.1.2.2	FFSSP formulation	146
8.1.2.3	Input dataset	149
8.1.2.4	Exact solution results	151
8.1.3	Metaheuristic solution	153
8.1.3.1	Tabu search model	153
8.1.3.2	Solution method	155
8.1.3.3	Metaheuristic solution results	157
8.2	Simulation of train discharge and loading	159
8.2.1	Physical components of the model	160
8.2.2	Logic behind the rail planner	162
8.2.3	Advanced rail dispatcher logic	163
8.2.4	Travel manager and parking spot selection strategy	164
8.2.5	Limitations and future work	165
Chapter 9	Rail Yard Capacity Analysis	166

9.1	Evaluation of theoretical rail yard capacity	167
9.1.1	General description of the rail park	167
9.1.2	Process modeling without locomotive constraints	169
9.1.2.1	Constraints without locomotive considerations	170
9.1.2.2	Results without locomotive constraints	176
9.1.3	Process modeling with locomotive constraints	179
9.1.3.1	Constraints with locomotive considerations	179
9.1.3.2	Results with locomotive constraints	183
9.2	Capacity evaluation through a simulation model	185
9.2.1	Structure and logic of the capacity model in FlexTerm	186
9.2.1.1	Model components	187
9.2.1.2	Connections between model components	188
9.2.1.3	Operational logic and rules	189
9.2.2	Simulation experiments based on the theoretical model	202
9.2.3	Simulation experiments based on the theoretical model with realistic elements added	207
9.2.4	Simulation experiments under model saturation	212
9.2.4.1	Changing the number of the RMGs	213
9.2.4.2	Changing operational constraints	221
Chapter 10	Conclusions and Future Directions	226
Apendix A		232
Bibliography		240

List of Figures

1	Throughput of containers over the years for Italy, Southern Europe, and Europe (Data source: [1])	19
2	Italy's Share of European container throughput over time (Data source: [1]) . . .	20
1.1	Global PSA network (Source: [2])	32
1.2	PSA GP main container areas	36
1.3	Main handling equipment operating at the PSA GP terminal.	39
3.1	PSA GP main components (Source: Google Earth)	53
3.2	BPMN diagram illustrating the truck arrival in the terminal for container discharge	55
3.3	BPMN diagram illustrating the housekeeping process for exporting containers . .	56
3.4	BPMN diagram illustrating the process of loading containers onto the ship	57
3.5	BPMN diagram illustrating train entry and exit in PSA GP	60
3.6	BPMN diagram illustrating the process flow of rail container discharge operations	61
5.1	Decision-support framework based on simulation modeling	85
5.2	Simulation environment structure with selected fixed resources	90
6.1	Conceptual framework of the research on the truck subsystem	99
6.2	Inbound container flow via trucks	100
6.3	Loading process of containers onto the vessel	101
6.4	Input masks and output relationship	104
6.5	Illustrative graph of time intervals for a ship i	107

6.6	Actual vs. ANN and Heuristic predicted container arrival curve (Ship 149)	110
6.7	Actual vs. ANN and Heuristic predicted container arrival curve (Ship 183)	111
6.8	Discharging process of containers from the vessel	112
6.9	Outbound container flow via trucks	112
6.10	Actual vs. ANN predicted TEU arrival curves for Dataset 1 (OF)	118
6.11	Actual vs. ANN predicted TEU arrival curves for Dataset 2	119
7.1	Container positioning and repositioning moves	127
7.2	Daily truck flow before and after redistribution	128
7.3	Hourly truck flow before and after redistribution (Day 13)	129
7.4	Hourly truck flow before and after redistribution	130
7.5	Actual vs. ANN predicted truck arrival curve for the period taken into consideration	131
7.6	Exhaust CO Hot Emissions before and after truck redistribution	136
7.7	Exhaust CO Hot Emissions before and after the first iteration of the redistribution algorithm	137
7.8	Exhaust CO Hot Emissions before and after the sixth iteration of the redistribu- tion algorithm	137
7.9	Exhaust CO Hot Emissions before and after optimization-based truck redistribu- tion ($E_{max} = 6492$)	140
7.10	Exhaust CO Hot Emissions before and after optimization-based truck redistribu- tion ($E_{max} = 5172$)	140
8.1	Conceptual framework of the research for rail discharge optimization	143
8.2	Main phases of the rail container discharging process: transfer from train to PM and from PM to yard stack	144
8.3	Solution performed by the terminal operators	152
8.4	Optimal solution of the proposed methodology	152
8.5	Scheduling of containers by terminal operators	158
8.6	Metaheuristic scheduling of containers (5 PMs)	159
8.7	Metaheuristic scheduling of containers (2 PMs)	159

8.8	Simulation model of the PSA GP terminal in FlexTerm	160
8.9	Main physical components for the modeling of rail operations in FlexTerm	162
9.1	Conceptual framework of the research for rail capacity evaluation	167
9.2	Railway park overview	168
9.3	Number of trains processed for each (N_K, N_G) pair, $N_O = 5$	177
9.4	Number of trains processed for each (N_O, N_G) pair, $N_K=2$	178
9.5	Number of trains processed for each (N_L, N_G) pair, $N_O = 5, N_K = 2$	184
9.6	Number of trains processed for each (N_L, N_G) pair, $N_O = 5, N_K = 2$	185
9.7	Capacity simulation model in Flex Term	187
9.8	Internal rail park	187
9.9	Distribution of containers for Train Type 1: (a) inbound trains and (b) outbound trains.	200
9.10	Distribution of containers for Train Type 2: (a) inbound trains and (b) outbound trains.	201
9.11	Number of trains handled within a 24-hour period under different simulation scenarios (Theoretical experiment)	206
9.12	Comparison of train movement Gantt charts under different locomotive avail- ability conditions (Theoretical experiment)	207
9.13	Number of trains handled within a 24-hour period under different simulation scenarios (Theoretical experiment with realistic operational parameters)	209
9.14	Trace Gantt of the trains showing the time trains spent in each of the elements of the model	210
9.15	Average stay time of trains across different track types under theoretical and realistic modeling assumptions	211
9.16	Number of trains handled over a 24-hour period under different RMG configura- tions	214
9.17	Resource utilization under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time	216
9.18	Resource activity over time under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time	217

9.19	Number of trains handled over one week under different RMG configurations . .	218
9.20	Resource utilization under saturated conditions for two representative scenarios with varying numbers of RMG cranes	219
9.21	Resource activity over time under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time over a one-week simulation horizon	220
9.22	Resource utilization under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time over a one-week simulation horizon	220
9.23	Number of trains handled over one week under different operational policies . . .	222
9.24	Resource activity over time under saturated conditions for two scenarios with different processing policies	223
9.25	Number of trains handled over one week under different operational policies under a relaxed processing policy	225
10.1	Distribution of containers for Train Type 3: (a) inbound trains and (b) outbound trains.	232
10.2	Distribution of containers for Train Type 4: (a) inbound trains and (b) outbound trains.	232
10.3	Distribution of containers for Train Type 5: (a) inbound trains and (b) outbound trains.	233
10.4	Distribution of containers for Train Type 6: (a) inbound trains and (b) outbound trains.	233
10.5	Distribution of containers for Train Type 7: (a) inbound trains and (b) outbound trains.	234
10.6	Distribution of containers for Train Type 8: (a) inbound trains and (b) outbound trains.	234
10.7	Distribution of containers for Train Type 9: (a) inbound trains and (b) outbound trains.	235
10.8	Distribution of containers for Train Type 10: (a) inbound trains and (b) outbound trains.	235
10.9	Distribution of containers for Train Type 11: (a) inbound trains and (b) outbound trains.	236

10.10	Distribution of containers for Train Type 12: (a) inbound trains and (b) outbound trains.	236
10.11	Distribution of containers for Train Type 13: (a) inbound trains and (b) outbound trains.	237
10.12	Distribution of containers for Train Type 14: (a) inbound trains and (b) outbound trains.	237
10.13	Distribution of containers for Train Type 15: (a) inbound trains and (b) outbound trains.	238
10.14	Distribution of containers for Train Type 16: (a) inbound trains and (b) outbound trains.	238
10.15	Distribution of containers for Train Type 18: (a) inbound trains and (b) outbound trains.	239
10.16	Distribution of containers for Train Type 19: (a) inbound trains and (b) outbound trains.	239

List of Tables

6.1	Input features for the export container prediction model	108
6.2	Statistics for each service (export containers)	109
6.3	Comparison of ANN and Heuristic results	110
6.4	Input features for the import container prediction model	116
6.5	Statistics for each service (import containers)	116
6.6	Predictive model results	117
6.7	A comparison with the Random Forest method	119
7.1	Main factors for EF_{hot} calculation	133
8.1	Processing times for each job on each machine (in seconds)	150
8.2	Set up times for each job on RMG (in seconds)	150
8.3	Set up times for each job on RS (in seconds)	151
9.1	Number of trains processed for each (N_K, N_G) combination over time horizons t = 1 to 24	177
9.2	Maximum number of trains processed for different values of N_L and N_G . The last two rows correspond to the dashed curves, i.e. the cases with unconstrained locomotives ($N_L = \infty$).	185
9.3	Train schedule in source	203
9.4	Key simulation parameters and selected configurations (Theoretical experiment) .	204
9.5	Experimental scenarios defined by the combination of <i>AvailabilityRate</i> and <i>time- ToAvailability</i>	204

9.6	Comparison between theoretical and realistic modeling assumptions	208
9.7	Key simulation parameters and selected configurations for the under-saturation model with different numbers of RMGs	213
9.8	Experimental scenarios for RMG configuration and locomotive return policy . . .	214
9.9	Number of completed trains over one week for different scenarios	221
9.10	Key simulation parameters and selected configurations for the under-saturation model under varying operational constraints	222
9.11	Operational policy scenarios under saturated conditions	222
9.12	Scenarios for the evaluation of processing flexibility through a Gray Interval . . .	224

Introduction

The first time I visited a port terminal, I was surprised by how many things were happening at once, cranes lifting containers, trucks moving in and out, vessels being discharged and loaded, yard workers coordinating on screens and radios. What looked like a simple goal, moving cargo from ship to shore and beyond, turned out to be a highly complex operation. Over time, I began to see the terminal not just as a single system, but as many independent systems working together. This idea closely matches the concept of a System of Systems (SoS), which [3] defines as a super system made up of separate, complex operational systems that interact to achieve a common goal. A port terminal brings together several distinct but interconnected subsystems, like yard management, berth allocation, gate operations, rail yard operations, and IT systems, each with its own complexity, autonomy, and responsibilities. Yet, they all need to work together efficiently to ensure that containers are handled smoothly and vessels, trucks and trains are served on time. From this perspective, it becomes clear that a port terminal can and should be viewed as a SoS, as also confirmed by [4]. This view has guided much of the thinking and direction behind the optimization work presented in this thesis.

Owing to its cost-efficiency and capacity to handle large-scale cargo, maritime transport continues to serve as a vital foundation of international commerce, as over 80% of world trade volume is carried by sea. In 2023, 11.6 billion metric tons of goods were loaded for international maritime trade, a slight decrease of 0.6% from 2022, and only 1.3% higher than in 2018. Although growth has slowed in recent years, international trade volumes have increased substantially over the long term. Since the start of the millennium, the total volume of goods loaded for international trade has grown significantly, with a 14.2% increase recorded between 2013 and 2018. During this period, dry cargo shipments, including containers, dry bulk, and general cargo, more than doubled [1].

As for the containerized port traffic, again according to [1], in 2023, global container ports processed around 858 million twenty-foot equivalent units (TEUs). Compared to 2013, when throughput was about 641 million TEUs, this represents a growth of 33% over the past decade. However, in both 2022 and 2023, container volumes increased by only 0.5%, marking the slowest pace of expansion in more than ten years, aside from the exceptional downturn during the COVID-19 pandemic in 2020. These trends underline the dual reality of containerized trade:

while the long-term trajectory remains one of significant growth, the recent slowdown emphasizes the need for ports to focus on improving efficiency and operational performance rather than relying solely on expanding traffic volumes.

Fig. 1 illustrates the trends in container throughput over time for Europe, Southern Europe, and Italy. Between 2010 and 2023, container throughput in Europe demonstrated a long-term growth trend, rising from approximately 86 million TEUs in 2010 to a peak of about 120.6 million TEUs in 2021, before declining to around 110 million TEUs in 2023. A similar pattern is observed in Southern Europe, where throughput grew from 28.2 million TEUs in 2010 to 43 million TEUs in 2021, followed by a slight contraction to 41.3 million TEUs in 2023. In contrast, Italy's container throughput has remained relatively stable, fluctuating between 9.7 and 11.5 million TEUs over the same period, with only modest growth compared to broader regional and continental trends. In 2023, Italy accounted for roughly 10% of Europe's total container traffic and about one-quarter of Southern Europe's throughput, underscoring its consistent but comparatively limited role in the region's containerized trade.

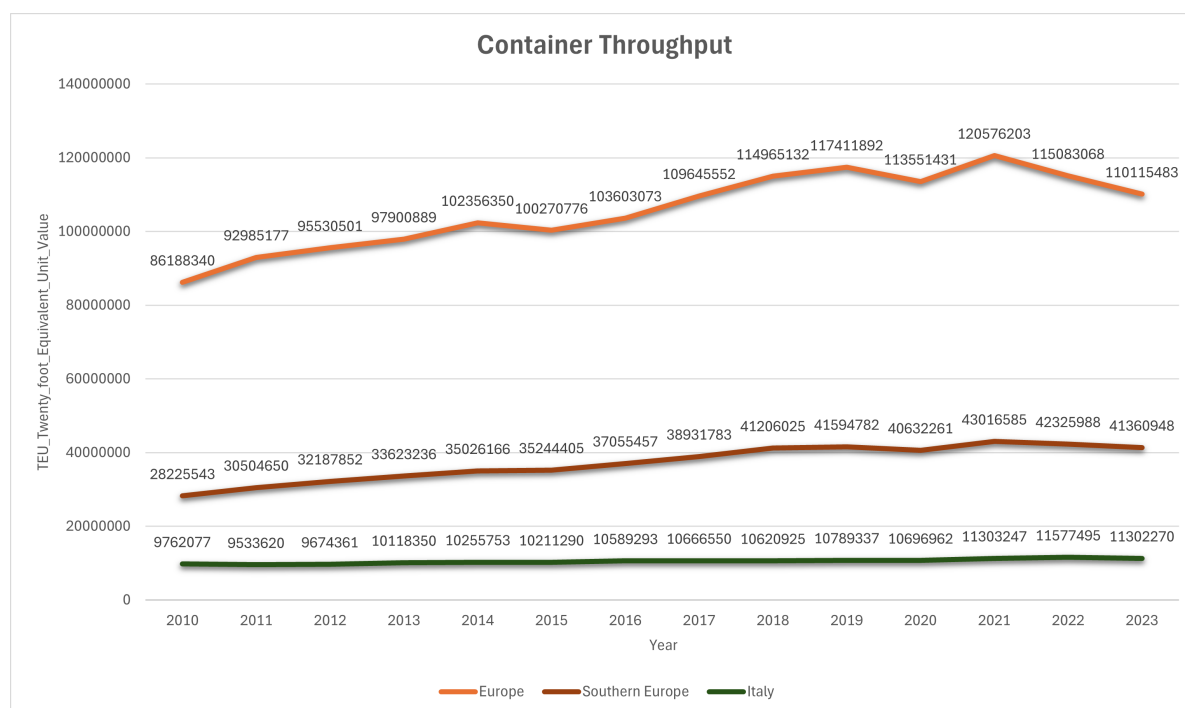


Figure 1: Throughput of containers over the years for Italy, Southern Europe, and Europe (Data source: [1])

Moreover, the graph in Fig. 2 shows how Italy's share of Europe's container throughput has evolved between 2010 and 2023. In 2010, Italy held a relatively high share of about 11.3%, but this quickly declined in 2011 to around 10.3%, and then fluctuated around 10-10.5% until

2016. From 2017 onwards, the trend reveals a gradual decline, reaching its lowest point in 2019 at approximately 9.2%, indicating a period where Italy was losing relative weight within the European container market. After 2019, Italy's share showed signs of stabilization and modest recovery, particularly from 2021 to 2023, when it rose again above 10%. By 2023, Italy's share stood at roughly 10.3%, suggesting that while the country did not return to the higher levels of the early 2010s, it regained some ground after the decline. These dynamics highlight the need for Italian ports to prioritize efficiency and process optimization, given that growth in volumes has remained relatively flat, unlike the stronger expansion trends seen at the European level.

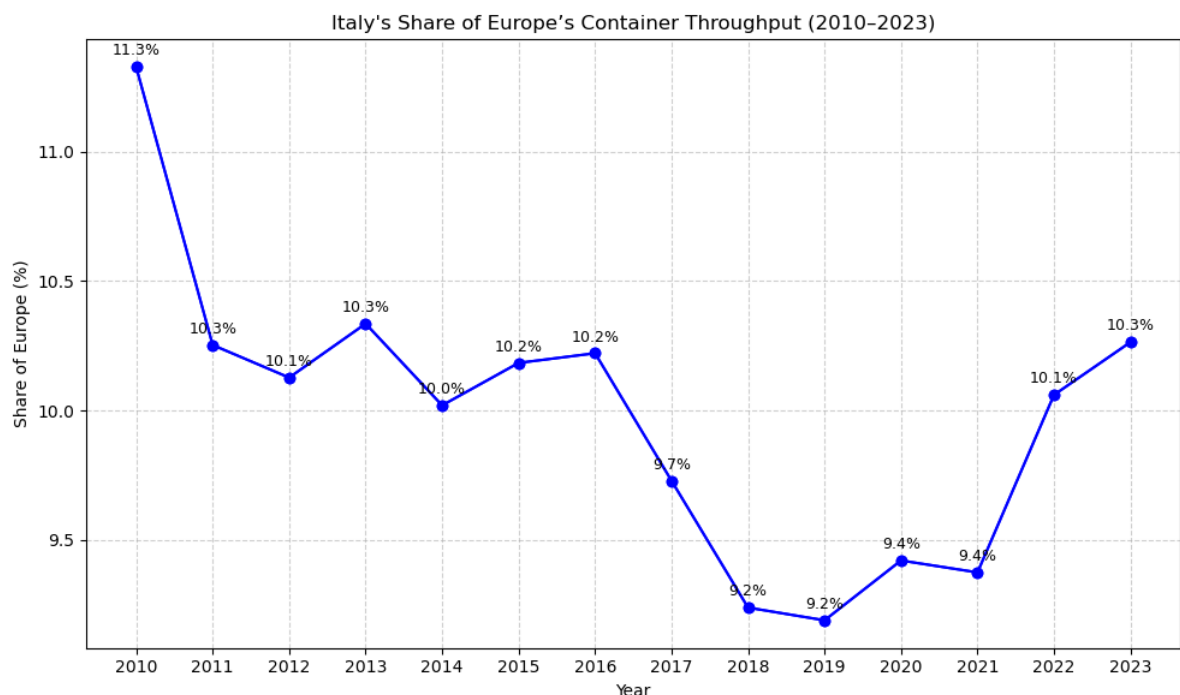


Figure 2: Italy's Share of European container throughput over time (Data source: [1])

To fully understand the implications of Italy's container throughput trends, it is essential to situate these national dynamics within the broader context of global maritime developments. While Italy has maintained comparatively stable volumes over the past decade, the wider logistics environment has become increasingly volatile and unpredictable. In recent years, ports worldwide have been confronted not only with slowing growth but also with severe external shocks that have reshaped shipping routes, vessel deployment strategies, and port demand. These global disturbances place additional operational demands on ports, including those in Italy, where stable throughput levels mean that competitiveness is increasingly supported by efficiency, reliability, and coordinated logistics rather than by expansion in scale.

To contextualize these increasing operational demands, it is useful to examine the major disruptions that have recently reshaped global maritime routes and global shipping patterns.

Between 2024 and 2026, multiple geopolitical crises targeted some of the most critical chokepoints in the world. In 2024-2025, escalating attacks in the Red Sea and Bab el-Mandeb Strait forced major carriers to abandon the area entirely, resulting in a more than 50% drop in Suez Canal traffic and forcing ships to reroute thousands of miles around the Cape of Good Hope and adding roughly 12 days to certain voyages [1].

Beyond the Red Sea and Suez Canal crisis, the Panama Canal also faced severe disruption in 2024, driven by historically low water levels that limited transit capacity. Traffic through the canal fell by more than 50% by mid-2024, forcing carriers to seek alternative routes and compounding the global rerouting already triggered by the Suez situation. These parallel disruptions, one driven by geopolitical conflict, the other by climate conditions, significantly lengthened global shipping distances, increased operational costs, and underscored how vulnerable key maritime chokepoints have become.

The situation intensified into 2025 as continued insecurity made normal passage impossible. By late 2023 and throughout 2024-2025, the Red Sea had effectively become a conflict zone, causing a 70% decline in ship tonnage through Suez and a corresponding 90% increase in traffic around southern Africa, according to [1] assessments. These disruptions strained global schedules, raised freight and insurance costs, and exposed the vulnerability of global supply chains that rely on predictable, uninterrupted maritime flows.

In 2026, the situation intensified further when military escalation in the Middle East prompted major shipping lines to suspend transit through the Strait of Hormuz, one of the world's most vital energy corridors [5]. These disruptions did not simply delay vessels, they reshaped trade flows, extended voyage times, raised fuel and insurance costs, reduced schedule reliability, and placed sudden, uneven pressure on port terminals worldwide [6]. As traffic surged at alternative ports and declined at others, terminals are forced to adjust to volatile, unpredictable cargo patterns.

Port terminals sit at the heart of global trade, absorbing the impact of every maritime disruption. As vessels reroute through longer, less predictable paths, terminals must handle fluctuating arrival patterns, irregular cargo surges, equipment imbalances, and shifting global trade corridors. For these reasons, optimizing port terminal operations is no longer optional. It is essential for:

- maintaining supply chain reliability in times of crisis;
- reducing congestion and vessel turnaround times;
- supporting resilience against geopolitical or climate-driven shocks;
- enabling more sustainable, fuel-efficient logistics networks;
- preventing minor disruptions from cascading into global shortages.

In a context where major logistics chokepoints can become inaccessible with little warning, the role of well-optimized, technologically advanced, and resilient port terminals has become

increasingly critical. These systems are essential for sustaining global trade flows and ensuring economic stability across regions.

Building on this perspective, the focus of the present research shifts toward the operational subsystems that most directly influence terminal efficiency and resilience. In particular, this thesis concentrates on two inland-facing components of a port terminal: the *gate system* and the *rail yard*. This focus is motivated by the fundamental role of hinterland connectivity, which determines how effectively a port integrates with external transport modes such as trucks, rail, and barges. Efficient inland connections reduce congestion, improve cargo flow, and enhance the terminal's ability to absorb operational disruptions. This further reinforces the concept of the port terminal as a *system of systems*, where multiple interdependent components must operate in a coordinated manner.

This work is developed as part of an industrial collaboration with a container terminal, from which the main research problems and practical challenges addressed in this thesis originate. The study is therefore driven not only by theoretical interest but also by the need to evaluate and improve real-world terminal operations.

The analysis focuses on the optimization of two key inland-side subsystems of the port terminal: the gate system, where trucks arrive to deliver or collect containers, and the rail yard, where trains are loaded and discharged. Although these subsystems are the primary focus of the study, they are examined in close connection with other terminal components, particularly the yard, where containers are stored and handled, and the berth, where vessels arrive and depart. This approach reflects the inherently interconnected nature of terminal operations, acknowledging that the performance of gate and rail processes cannot be fully understood without considering their dependence on yard availability and vessel-related activities. The objective is not to analyze these components in isolation, but rather to understand their combined impact on overall terminal operations.

The literature on port terminal optimization is extensive, with a large body of studies addressing a wide range of operational problems. However, the majority of this research has traditionally focused on the quay side of the terminal, where vessels are berthed and served. In particular, significant attention has been devoted to the Berth Allocation Problem (BAP), which concerns the optimal assignment of vessels to berth positions along the quay. This problem depends on several factors, including berth length and depth, vessel dimensions, arrival times, and service windows, and has been addressed using various modeling approaches [7, 8, 9, 10, 11].

Closely related to berth allocation, the Quay Crane Scheduling Problem (QCSP) has also been widely studied in the literature [12, 13]. This problem focuses on determining the optimal assignment and sequencing of quay cranes for loading and unloading vessel containers, accounting for operational constraints such as crane interference, safety distances, and workload balancing. Given its direct impact on vessel turnaround time and terminal productivity, QCSP has been approached using both classical optimization techniques, including mathematical programming

and heuristic methods [14], as well as more advanced approaches such as reinforcement learning and hybrid intelligent frameworks [15, 16].

In addition, significant attention has been devoted to yard management problems, including container storage strategies, reshuffling minimization, resource allocation, and coordination of handling equipment [17, 18, 19]. Another well-studied area is ship stowage planning, a highly constrained problem where loading decisions must comply with vessel-specific requirements and are subject to validation by the ship’s captain [20, 21].

Despite the extensive research on these core subsystems, less attention has been given to inland operations, particularly those related to rail processes. While some studies address train loading and discharge operations [22, 23], there remains limited understanding of how rail terminal capacity and performance influence overall terminal operations, especially when considered alongside other subsystems. This limitation is particularly relevant given the increasing importance of rail transport in sustainable logistics systems. Efficient rail operations have the potential to reduce congestion and emissions at a system level, underscoring the need to better understand their role within port terminal operations.

Similarly, although truck arrival prediction and the design of Truck Appointment Systems (TAS) have been widely studied and provide valuable insights into gate-level performance [24, 25], these approaches often consider truck flows in isolation, without explicitly accounting for their interactions with other terminal subsystems. As a result, limited attention has been given to the integration of truck-related decision-support tools with yard, berth, and rail operations. In particular, the potential of arrival prediction models and appointment systems to support coordinated terminal-wide optimization, such as mitigating yard congestion, improving synchronization with vessel and rail schedules, or reducing emissions through smoother traffic flows, remains insufficiently explored. This highlights the need for more comprehensive approaches that embed truck-related processes within a system-level view of port terminal operations.

As an industrial PhD thesis, this work adopts a dual perspective. From an applied standpoint, it aims to support the partner terminal by analyzing and improving specific operational processes through data-driven and model-based decision-support approaches. From an academic standpoint, it seeks to contribute to the literature by addressing the limited system-level understanding of rail yard capacity and operations within port terminals, together with a systematic analysis of gate-side dynamics and their traffic-related and environmental implications.

Building on the challenges and research gaps identified above, this thesis focuses on inland-side port terminal operations, with particular emphasis on rail yard capacity, rail yard operations, and gate processes. The scope of the work is centered on understanding how these components affect terminal performance under conditions of operational variability, congestion, and sustainability constraints. Although the study is developed in collaboration with an industrial partner, the analysis is framed to generate insights that are applicable to port terminal operations more generally.

In close collaboration with the partner terminal, the objectives of this thesis are as follows:

- To analyze how gate operations, particularly truck arrival patterns, are influenced by vessel arrivals and berth-related activities, and how these relationships affect inland-side terminal performance.
- To evaluate the role of gate-related processes in contributing to yard congestion and overall terminal efficiency, with specific attention to the implications of truck arrival dynamics.
- To investigate the environmental impact of gate operations, focusing on emissions associated with truck arrivals and movements within the terminal.
- To analyze the operational performance of rail yard processes, with particular emphasis on train discharge activities within the port terminal.
- To assess rail yard operations through a detailed simulation-based representation of internal processes, in order to better understand their influence on overall terminal performance.
- To analyze rail yard capacity as a distinct aspect of terminal performance, considering both analytical and simulation-based perspectives.
- To support terminal-level decision-making by evaluating alternative inland-side operational configurations and scenarios.

Based on the objectives defined above, the research is guided by the following research questions:

- How can vessel-related and berth-side operational information support the analysis and prediction of truck arrival patterns at the terminal gate?
- How do gate operations and truck arrival dynamics contribute to yard congestion and inland-side terminal performance?
- What are the environmental impacts associated with gate-related truck movements within the terminal?
- How do rail yard discharge operations affect inland-side terminal performance and resource utilization?
- How can rail discharge scheduling operations be improved through the application of analytical and optimization-based approaches?
- How do internal rail yard processes influence terminal performance when analyzed through a detailed simulation-based representation?

- What are the capacity limitations of the rail yard, and how can these be evaluated through analytical and simulation-based methods?
- How can scenario-based analysis support decision-making for inland-side port terminal operations?

Building on the objectives and research questions outlined above, this thesis provides a set of contributions addressing both practical and academic challenges related to inland-side port terminal operations. The main contributions are summarized as follows.

Building on the objectives and research questions outlined above, this thesis provides a set of contributions addressing both practical and academic challenges related to inland-side port terminal operations. The main contributions are summarized as follows.

- An analysis of truck arrival dynamics at the terminal gate, examined in relation to vessel arrivals and berth-side activities, in order to evaluate their impact on yard congestion and inland operational performance.
- An evaluation of the traffic-related and environmental impacts associated with gate operations, with particular focus on truck-generated emissions and congestion effects within the terminal area.
- The development and application of optimization approaches for rail yard discharge scheduling problems, aimed at improving resource utilization and operational efficiency.
- The design and implementation of a detailed simulation model for rail discharge operations, aimed at reproducing the operational behavior and interactions of terminal resources during train handling activities.
- The development of a mathematical framework for rail yard capacity evaluation, enabling the estimation of theoretical capacity limits.
- The development of a simulation-based framework for evaluating rail yard capacity under different operational scenarios and resource configurations.

The remainder of this thesis is organized as follows. Chapter 1 provides an overview of port terminals, introducing their main components and operational processes, and outlining the motivations for conceptualizing a port terminal as a system of systems. This chapter also presents the partner port terminal with which the thesis has been developed in close collaboration.

Chapter 2 reviews the relevant literature related to the main research topics addressed in this thesis, with particular emphasis on port terminal operations and inland-side processes.

Chapter 3 focuses on the case study terminal and describes its system architecture from a system-of-systems perspective. The chapter details the main real-world operational processes occurring within the terminal, which form the basis for the models and analytical approaches developed in the subsequent chapters.

Chapter 4 presents the theoretical foundations of the methodologies and analytical approaches adopted to address the research objectives of the thesis. Given the central role of simulation in this work, Chapter 5 is dedicated specifically to simulation-based approaches, providing a detailed discussion of the underlying concepts, motivations, and the simulation tools employed. This separation reflects the importance of simulation within the overall methodological framework of the thesis.

The subsequent chapters present the core developments and contributions of the research. Chapter 6 introduces the predictive analyses related to container flows handled by trucks, addressing both import and export operations, and discusses the results obtained through the proposed methodologies. Building on these results, Chapter 7 examines how predictive insights can be leveraged to optimize terminal operations, mitigate yard congestion, and assess traffic-related and environmental impacts, including the reduction of emissions.

Chapter 8 addresses the second main inland-side subsystem considered in this thesis, focusing on the improvement of rail yard operations, with particular attention to train discharge processes.

Chapter 9 subsequently investigates the problem of rail yard capacity evaluation, approached through both analytical methods and, more extensively, through simulation-based modeling and scenario analysis.

Finally, Chapter 10 concludes the thesis by summarizing the main findings and contributions of the research and by outlining potential directions for future work.

For the purposes of this thesis, the terms “vessel” and “ship” are used interchangeably.

Chapter 1

Introduction to Port Terminals

Port terminals represent fundamental components of the global transportation and logistics network. They serve as critical interfaces between maritime transportation and inland transport systems, enabling the transfer of goods between ships and land-based modes such as rail, road, and inland waterways. Through these functions, port terminals play a key role in facilitating international trade and supporting the efficient movement of cargo across global supply chains. Beyond their role as simple transfer points, modern port terminals perform a wide range of operations including cargo handling, temporary storage, consolidation and distribution of goods, and coordination of different transport flows. These activities require the integration of infrastructure, specialized equipment, operational procedures, and information systems to ensure that cargo can be processed efficiently and reliably.

Beyond their central role in sustaining global trade, the importance of port terminals also lies in their broader economic and social contributions. Globally, ports support millions of jobs, both directly ,involving dockworkers, terminal operators, and logistics professionals, and indirectly through related sectors such as manufacturing, warehousing, and inland transport. Their capacity to facilitate exports, attract foreign investment, and strengthen supply chain resilience positions them as strategic assets for national and regional development.

As a result, port terminals can no longer be viewed simply as physical infrastructures, but rather as complex operational environments where multiple processes, resources, and transport modes interact. Understanding how these different elements interact is essential for analyzing terminal performance, identifying bottlenecks, and improving operational efficiency. For this reason, port terminals can be studied as complex systems composed of multiple interconnected subsystems, each responsible for specific operational functions within the terminal.

This chapter first presents an overview of the main components of a port terminal. The focus is then directed to the port terminal that serves as the case study of this thesis and with which this research was carried out in collaboration.

1.1 The terminal as a system-of-systems

A port terminal, due to its operational complexity and the inherent intricacy of the wider maritime port environment, can be classified as a System of Systems (SoS). Its functioning depends on the interaction and coordination of multiple interconnected subsystems, each fulfilling a distinct yet complementary role within the overall logistics chain.

From an operational perspective, a port terminal can be viewed as an integrated ecosystem composed of several key subsystems. The most central components are the berth area, where vessels are moored and cargo is transferred between ship and shore, and the yard, which serves as the temporary storage area for containers awaiting further movement.

In addition to these core areas, other critical operational subsystems support the overall functioning of the terminal. These include the gate, where external trucks enter and exit to deliver or collect containers, and the rail yard, which performs a similar role for intermodal transport by train. Together, these elements ensure the smooth interface between maritime and inland transportation flows, forming a continuous logistics cycle.

1.1.1 Berth operations

Within a container terminal, the berth represents one of the core subsystems in the overall system of systems that defines terminal operations. It is the area where vessels are located and where the discharge and loading of containers take place, linking the maritime domain with the landside logistics network. The performance of berth operations has a direct influence on the overall productivity of the terminal, as it determines vessel turnaround times and affects the utilization of equipment and resources across the entire system. Berth operations are often coordinated through a Terminal Operating System (TOS) or other supporting software systems, enabling the management of vessel schedules, berth allocation, quay crane assignments, and integration with yard and landside activities.

Berths are typically equipped with quay cranes, which are specialized handling units designed to transfer containers between vessels and the yard equipment. The coordination between quay cranes, yard equipment, and landside transport modes makes the berth a critical interface where multiple subsystems interact.

From an operational point of view, three main container handling cycles can be identified:

- *Export cycle:* containers arrive at the terminal via land transport, such as truck or rail, and are subsequently loaded onto vessels for maritime transport to their final destinations.
- *Import cycle:* containers are discharged from vessels and then delivered to their inland destinations through the terminal's road or rail connections.

- *Transshipment cycle*: containers are discharged from one vessel and loaded onto another, allowing for redistribution along different maritime routes.

In addition to these sea-land interactions, intermodal cycles also may exist within a terminal, in which containers arrive and depart exclusively via land transport, such as truck-to-rail, rail-to-truck, truck-to-truck, or rail-to-rail movements. Although this process does not involve the berth directly, it remains an integral part of the overall terminal system, contributing to the efficient transfer of cargo between different transport modes. The effective integration of these operational cycles, together with the coordination between the berth, yard, gate, and rail subsystems, is essential for achieving high terminal performance and ensuring a smooth flow of containers throughout the entire logistics chain.

1.1.2 Gate operations

The gate constitutes one of the principal interface between the terminal and the external road transport network. It is the subsystem through which trucks enter and leave the terminal to deliver or collect containers, linking inland logistics flows with the internal operational environment. In a system-of-systems view, the gate acts as the physical and informational boundary between the terminal and its hinterland.

Gate operations consist of a sequence of control and verification activities designed to ensure the secure and efficient movement of cargo and vehicles. These typically include the identification of vehicles and drivers, the validation of booking information, the inspection of documentation, and, when required, customs clearance procedures. Continuous coordination with the yard subsystem is necessary so that containers can be positioned in advance for pickup or delivery, avoiding congestion and idle time.

Modern terminals employ a high degree of automation at the gate. Technologies such as optical character recognition (OCR), automatic number-plate recognition, appointment scheduling, and self-service kiosks enable rapid processing of vehicles and reduce waiting times. The gate is also tightly integrated with the TOS and external logistics platforms, which synchronize the exchange of operational and administrative data in real time.

As a subsystem, the gate plays a crucial role in maintaining the balance between the landside transport network and internal terminal operations. Its performance affects yard productivity, berth planning, and the reliability of intermodal connections. Efficient gate management is therefore essential for the overall optimization of container flows across the entire terminal system.

1.1.3 Rail operations

The rail subsystem provides the main interface between the terminal and the external rail freight network, enabling the efficient movement of containers over medium and long distances. It plays a fundamental role in supporting intermodal transport and reducing road congestion, offering a sustainable alternative for the inland distribution of cargo. Within the terminal's system of systems, the rail operations interact closely with the yard, gate, and berth subsystems, forming an essential link between maritime and inland logistics flows.

Rail operations are typically organized around a dedicated rail yard equipped with multiple tracks and specialized handling equipment such as rail-mounted gantry cranes (RMGs). These units transfer containers between rail wagons and the terminal yard, where containers are either temporarily stored or prepared for loading onto vessels. Efficient coordination between train schedules, yard planning, and berth activities is necessary to ensure a balanced flow of containers and to minimize dwell times for both rail and maritime operations.

From a systemic perspective, the rail subsystem contributes to the overall performance of the terminal not only through physical operations but also through informational integration. Real-time data exchange between the TOS and railway operators allows accurate planning of train arrivals and departures, allocation of handling equipment, and synchronization with vessel schedules. Such coordination ensures that rail operations are effectively embedded within the terminal's broader logistics processes.

The rail subsystem therefore acts as a key enabler of intermodal connectivity, enhancing the terminal's capacity to serve distant hinterland markets while supporting environmental and efficiency objectives. Its seamless integration with other subsystems, particularly the yard and gate, is essential to achieving high levels of operational reliability and optimizing container flows across the entire transport chain.

1.1.4 Yard operations

The yard constitutes the central subsystem within a container terminal, acting as the operational hub that connects the maritime side with the landside transport systems. It functions primarily as a temporary storage and transfer area where containers are positioned, sorted, and prepared for further movement to or from other subsystems such as the berth, gate, or rail terminal. Within the system-of-systems framework, the yard enables the coordination of container flows between these interconnected subsystems, ensuring operational continuity across the entire terminal.

Yard organization typically follows a structured layout based on both operational function and cargo characteristics. Dedicated areas are assigned for import, export, and empty containers, while additional specialized zones accommodate refrigerated containers (reefers) and dangerous goods (IMO-classified cargo). Reefer containers require power supply points and constant

monitoring to maintain controlled temperatures, whereas IMO containers are stored in strictly regulated zones equipped with containment and fire protection systems. The segregation of these areas is essential for safety, accessibility, and compliance with international standards.

Containers are generally stored in stacks, and their arrangement is determined by several operational criteria, such as destination port, vessel service, weight, size, and expected dwell time. Proper stacking strategies are crucial to minimize rehandling operations and to facilitate efficient container retrieval when required for vessel loading or inland delivery. Stacking policies also vary depending on the level of yard automation and the type of handling equipment employed.

Different types of handling equipment can be used in yard operations, depending on the terminal's layout and level of mechanization. Rubber-tired gantry cranes (RTGs) are widely adopted for their flexibility and mobility, allowing container handling across multiple stacking rows. RMG cranes are typically used in automated or semi-automated yards, providing high precision and efficiency along fixed rails. Straddle carriers and automated stacking cranes (ASCs) allow independent container transport and stacking, offering high operational speed in large, high-throughput terminals. Reach stackers and front loaders, on the other hand, provide greater versatility for smaller terminals or specific operational areas, although they generally result in lower stacking density and higher rehandling rates compared to gantry-based systems.

Yard operations therefore involve a continuous interaction between planning, equipment deployment, and information management. Modern terminals rely on a TOS to monitor container locations, manage stacking orders, and allocate handling resources in real time. Through this digital layer, the yard subsystem interacts dynamically with the berth, gate, and rail systems, ensuring synchronized flows of containers and efficient utilization of available capacity.

In essence, the yard acts as the heart of terminal operations, balancing incoming and outgoing flows between maritime and inland transport. Its organization, handling equipment, and control strategies directly influence the overall performance, reliability, and sustainability of the terminal as an integrated system.

1.2 PSA International

PSA International is one of the leading global port operators and supply-chain solution providers. Headquartered in Singapore, the company manages and develops a worldwide portfolio of port terminals, logistics platforms, and related maritime services. Its operations extend across multiple regions, including Southeast Asia, Northeast Asia, the Middle East and South Asia, Europe and the Mediterranean, and the Americas.

The organization originated from the former Port of Singapore Authority, which was established in 1964 to regulate and operate Singapore's port. Over time, the institution evolved from a national port authority into a global port operator. In 2003, PSA International was established

as the investment holding company responsible for coordinating the group's worldwide port and supply-chain activities.

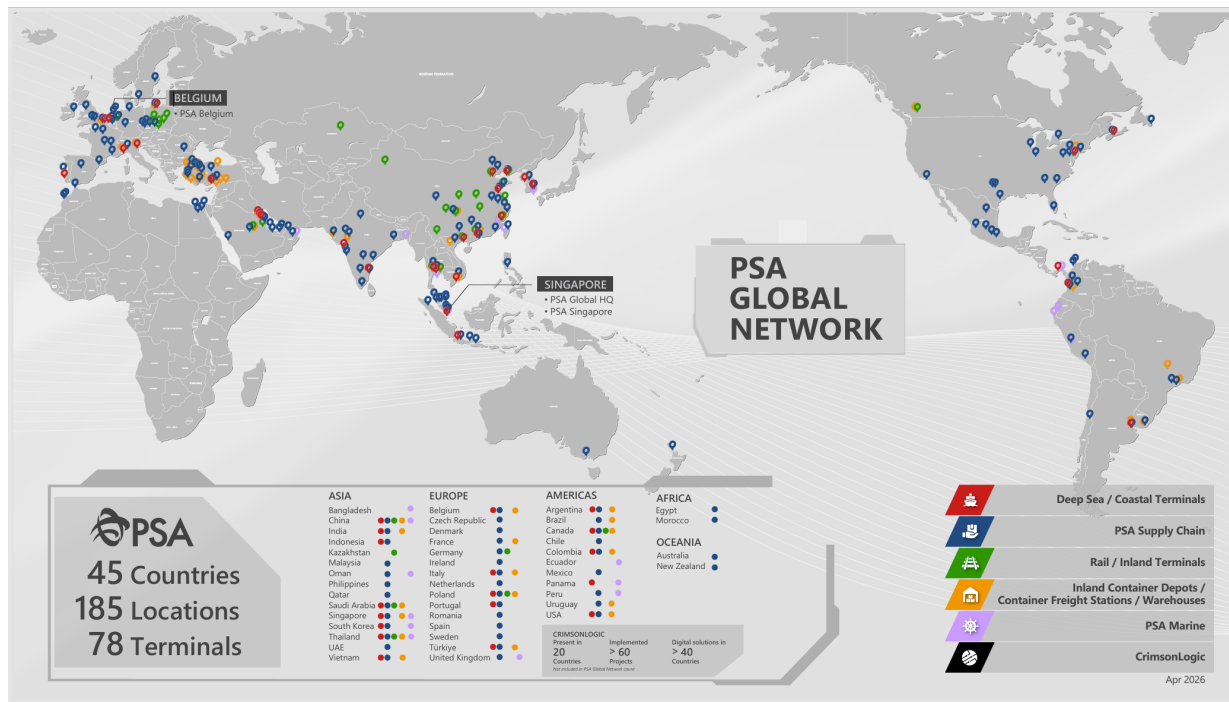


Figure 1.1: Global PSA network (Source: [2])

Today, PSA operates an extensive global network that includes approximately 185 locations in more than 40 countries, as illustrated in Fig. 1.1. This network comprises around 78 terminals, including deep-sea terminals as well as rail and inland terminals. Through this global infrastructure, PSA supports international trade by providing port operations, cargo handling services, and integrated supply-chain solutions that connect maritime transportation with inland logistics systems. In addition to port operations, the group is also involved in related activities such as logistics management, maritime services, and digital solutions.

The operational entities of PSA are organized into five major business areas known as “Regions”: Southeast Asia, the Middle East and South Asia, Northeast Asia, Europe and the Mediterranean, and the Americas. Each region is managed by a Regional Chief Executive Officer responsible for the operational and financial performance of the terminals within that region. The Regional CEOs report directly to the Group CEO, who is supported by a senior management team based at the PSA Corporate Center.

Currently, PSA employs approximately 30,000 people worldwide, including around 20,000 direct employees. The company's main hub is located in Singapore, which represents the largest

transshipment hub in the world. The Singapore hub alone handles nearly one-seventh of global transshipment container traffic and in 2021 handled approximately 37.2 million TEUs.

One of the most important terminals within PSA's international network is located in the Port of Antwerp. Through PSA Antwerp, the group operates three major container terminals, making it one of the most significant port operators in Belgium. This investment represents the largest PSA project outside Singapore. The port has a handling capacity of approximately 9 million TEUs and is equipped with 38 quay cranes and 228 yard cranes. Furthermore, with nearly 3,500 reefer slots available, it hosts one of the largest refrigerated container storage areas in Europe. Due to its strategic importance, PSA Antwerp serves as the headquarters of the Europe, Mediterranean, and Americas region [2].

Within this regional structure, PSA also operates several container terminals across Europe, including facilities located in major Mediterranean and Northern European ports. Among these terminals is the PSA Genoa Pra' terminal, located in the Port of Genoa, which represents the case study analyzed in this thesis.

1.3 PSA Genova Pra' terminal

As already mentioned in previous sections, the main case study on which this research is based is the PSA Genova Pra' (PSA GP) terminal. The objectives of this thesis concern the optimization of the terminal's operational processes. This section provides a general overview of the PSA GP terminal in order to contextualize the analytical and optimization work developed throughout this thesis. The description includes a brief review of the terminal's history, its geographical location and strategic role, as well as its infrastructure, capacity, main operational areas, and key handling equipment. This background information serves to facilitate a better understanding of the operational framework within which the proposed methodologies and analyses are applied. The information presented in this section is mostly based on data available on the official PSA GP website [26]

1.3.1 History

The origins of the port area of Pra' date back to a project developed during the 1970s and 1980s by the former Consorzio Autonomo del Porto di Genova, which is now part of the Western Ligurian Sea Port Authority (Autorità di Sistema Portuale del Mar Ligure Occidentale). The project aimed to expand the capacity of the Port of Genoa and accommodate the growing demand for containerized cargo.

In 1992, FIAT Impresit established SINPORT, a port services company, and obtained a concession from the Port Authority to complete and manage the container terminal of Pra'. During the

same period, the terminal began its operational development, with the first freight ferry calling in 1992, the first car carrier in 1993, and the first container vessel on May 5, 1994.

Since 1998, the terminal has been part of PSA International, the global operator introduced in the previous section. Under PSA's ownership, PSA GP has experienced continuous growth and modernization. Despite the economic downturn of 2008, the terminal maintained steady operations and contributed to the Port of Genoa surpassing the milestone of two million TEU handled annually. In 2015, a major modernization program was launched to improve efficiency, safety, and sustainability, positioning PSA GP as the first and only Italian gateway terminal capable of accommodating Ultra Large Container Vessels (ULCVs) of up to 20,000 TEU.

On 6 August 2020, PSA Genoa Investments NV, a Belgian company 62% owned by PSA International, acquired control of both PSA Genova Pra' S.p.A. and Terminal Contenitori Porto di Genova S.p.A. (PSA SECH), with the goal of creating operational synergies and consolidating PSA's leadership in the Northern Tyrrhenian area.

1.3.2 Location and strategic role

PSA GP is located in the western part of the Port of Genoa and serves as a key gateway to the Italian market, as well as to Central and Southern Europe. Thanks to its strategic geographic position, the terminal offers significantly shorter transit times, up to four days less, for containers arriving from East–West maritime routes compared to ports in Northern Europe.

Because of this advantageous location, PSA GP has become an essential stop along the main global shipping routes and is the preferred terminal for the world's leading liner alliances, including The Alliance, Ocean Alliance, and 2M. Since the establishment of these consortia in 2015, the terminal has consistently served as a reference point for major shipping companies.

The terminal also functions as a strategic intermodal hub, offering both rail and road connections that ensure efficient cargo distribution across Europe. Its rail park provides regular intermodal services to major industrial and commercial areas in Italy, Switzerland, Germany, and other parts of continental Europe. A notable development was the inauguration of the Mittelmeerzug (MMZ) service at the end of 2018, a direct rail link between Genoa and Basel. This connection reduces transit times by at least seven days compared to maritime routes to Northern Europe and provides a reliable and sustainable alternative to inland waterway transport. The train operates three round trips per week, carrying both high-cube containers and ADR-classified cargo, with a total length of 440 meters and a capacity of approximately 1,250 tonnes. Road access to the terminal is equally efficient. PSA GP is directly connected to the Pra' motorway junction, allowing trucks to bypass urban traffic and quickly access the main motorway network, both towards La Spezia (A10) and Milan (A26). These multimodal connections reinforce the terminal's role as a major logistics hub and as a critical link between maritime trade routes and inland European markets.

1.3.3 Infrastructure and capacity

PSA GP covers a total operational area of approximately 978,000 square metres, divided into six modules. The terminal's yard forms the core of its operations, where container handling and storage activities take place. The yard includes around 14,500 ground slots, 978 reefer plugs, and 320 dedicated ground slots for dangerous goods. The facility is equipped with a VGM weighing station, in compliance with the SOLAS Convention, and operates with electric rubber-tyred gantry cranes (E-RTGs) that ensure high performance with minimal environmental impact.

The IMO area provides safe handling and temporary storage for hazardous cargo. It has a maximum geometric capacity of 2,004 TEU, divided according to IMO classes, and is equipped with sealed containment basins for recovery and disposal of potential leakages (for classes 6, 8, and 3). The area is surrounded by a fire-protection ring, featuring an auxiliary pumping station and dedicated extinguishing equipment (for classes 2, 4, 9, and 5). In compliance with local regulations, the facility operates under the Genoa Port Authority's integrated fire-safety ordinance (No. 61/2001). The dangerous-goods yard is continuously monitored and supported by a dedicated fire-prevention service.

For temperature-controlled cargo, PSA GP provides 978 reefer plugs and performs two inspections daily to ensure proper functioning and cargo safety.

The berth extends for 1,675 metres and is accessible through a wide navigation channel that allows safe manoeuvring for all vessel types, including the latest-generation ULCVs of up to 20,000 TEU capacity. The basin layout enables the simultaneous berthing of three ULCVs, which are served by new gooseneck quay cranes designed to reach across large-scale vessels. The terminal's lighting and natural draught permit safe operations even during night-time navigation.

The gate complex supports efficient truck operations with 12 inbound and outbound lanes, including two bidirectional and interchangeable tracks. The system is equipped with automated gate-in and gate-out processes using OCR technology for both empty and loaded trailers, and with two dedicated lanes for non-containerized or exceptional cargo. The gate can handle more than 2,400 vehicles per day and is directly connected to the Pra' motorway junction, minimizing interference with urban traffic. The truck service area includes 10 automated self-service totems for registration and 10 staffed counters for document verification and acceptance, supported by semi-automated IT systems and mobile-app integration for drivers. The documentation and cargo acceptance office operates until 20:50 from Monday to Friday and 12:50 on Saturdays.

PSA GP also operates a rail terminal directly connected to the national railway network and dedicated freight lines. The facility includes eight internal tracks, each 900 metres long, and is equipped with four RMG cranes, enabling the handling of up to 120 trains per week. The rail operations run continuously from 06:00 on Monday to 06:00 on Sunday, with 24-hour train loading and unloading teams. The terminal accommodates trains up to 440 metres in length and can handle containerized dangerous goods. PSA GP also acts as a Multimodal Transport Operator

(MTO), managing complete rail shipments to and from northern Switzerland, particularly Basel, and providing intermodal connections to destinations in Northern Europe, including Rotterdam, Neuss, and Wolfurt (Austria).

1.3.4 Main container areas

In this section, the main operational areas of the PSA GP terminal are described in relation to the positioning and type of containers handled within the different operational cycles. Each area is identified by a specific color in the terminal layout, as illustrated in the following figure (Fig. 1.2) .

Container flows within the terminal can be broadly classified into import, export, transshipment, and intra-modal movements, as described in previous sections. These different cycles determine the utilization of terminal areas and the interaction among subsystems.

In addition to these main operational cycles, housekeeping activities are performed to reorganize containers within the yard when initial storage positions are not optimal, supporting efficient terminal operations across all container flows.

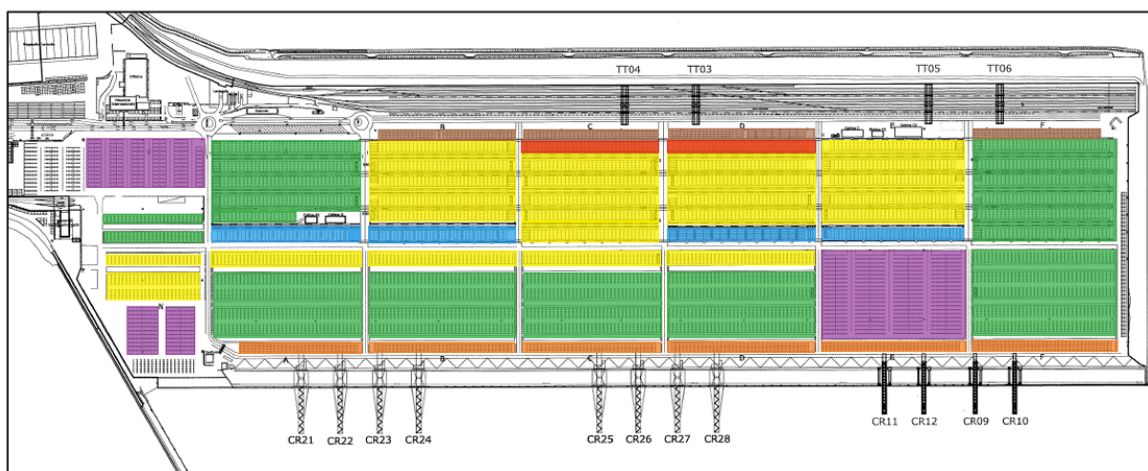


Figure 1.2: PSA GP main container areas

- **Export Area (Green):** This area is used to store containers that will subsequently be loaded onto vessels. The majority of these containers enter the terminal by truck, while the remaining portion arrives by vessel (transshipment containers) or by train.
- **Import Area (Yellow):** This area stores all containers that arrive at the terminal by vessel and will subsequently leave the terminal by truck.

- **Empty Container Area (Purple):** This area is dedicated exclusively to empty containers, which mainly arrive by vessel and, to a lesser extent, by train. These containers may exit the terminal by vessel, truck, or train.
- **Reefer Area (Light Blue):** This area is equipped with electrical power outlets and is dedicated to the storage and connection of refrigerated containers arriving by vessel or by truck.
- **Out-of-Gauge Area (Orange):** This area stores containers that exceed the standard dimensions of a container. These typically include industrial machinery, vehicles, or other heavy industry components transported on flat-rack containers. These areas are generally located close to the quay to facilitate storage and subsequent vessel loading operations.
- **Rail Area (Brown):** This area stores containers arriving by vessel or by truck that will subsequently be loaded onto trains operating within the two rail yards of the terminal.
- **IMO Area (Red):** This area is exclusively dedicated to containers classified as dangerous goods according to international maritime transport regulations. These areas are equipped with dedicated containment basins for the collection of hazardous substances in the event of accidental spills.

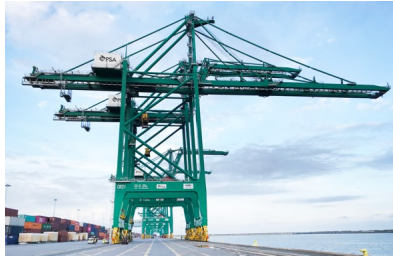
1.3.5 Handling equipment

Several types of handling equipment operate within a container terminal, each characterized by a specific operational role. This section describes the main handling equipment used at the PSA GP terminal, highlighting their primary functions as well as their main strengths and limitations within terminal operations.

- **Quay Cranes (QC):** Quay Cranes are used to load and unload containers from vessels berthed at the quay. PSA GP operates different types of quay cranes, capable of serving both large container vessels and feeder ships. This allows the terminal to handle vessels of different sizes simultaneously. All quay cranes are equipped with OCR systems that automatically identify container numbers and International Organization for Standardization (ISO) codes, thereby eliminating the need for manual inspection.
- **Rail-Mounted Gantry Cranes (RMG):** RMG Cranes, also referred to as TransTainers, are the main equipment used for train loading and unloading operations. They operate on dedicated rails along the rail yards and are electrically powered. Their configuration allows direct container handling between rail tracks and adjacent yard lanes, enabling efficient and autonomous rail operations.

- **Rubber-Tired Gantry Cranes (RTG):** RTGs are one of the primary yard handling equipment used for container stacking operations. At PSA GP, both diesel-powered and electric-powered RTGs are in operation, with a progressive transition toward electric units. RTGs allow efficient space utilization within the yard by stacking containers in multiple tiers, although their large dimensions limit their flexibility and relocation speed.
- **Reach Stackers (RS):** RSs are characterized by high operational flexibility and are capable of rapid relocation within the terminal. They are mainly employed in export operations, truck receiving activities, rail-related movements, and the handling of special containers. Their main limitation lies in the higher number of reshuffling movements and higher fuel consumption compared to gantry cranes when handling similar container volumes.
- **Prime Movers (PM):** PMs are the main vehicles used for horizontal container transportation within the terminal. They operate across all terminal cycles, transporting containers between quay cranes, yard areas, rail yards, and gates, and primarily serve lifting equipment rather than performing lifting operations themselves.
- **Front Loaders (FL):** FLs are primarily used for operations involving empty containers. Their compact structure and lifting configuration make them suitable for high-density stacking and handling of empty containers within dedicated yard areas. Due to the lower weight of empty containers, Front Loaders provide an efficient and cost-effective solution for empty container storage and repositioning activities.

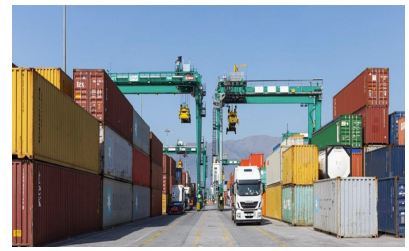
Fig. 1.3 illustrates the main handling equipment operating at the PSA GP terminal, as described in this section.



(a) Quay Crane



(b) Rail-Mounted Gantry Crane



(c) Rubber-Tired Gantry Crane



(d) Reach Stacker



(e) Front Loader



(f) Prime Mover

Figure 1.3: Main handling equipment operating at the PSA GP terminal.

1.3.6 Key challenges in PSA GP terminal operations

Port terminals are complex systems composed of multiple interconnected subsystems and logistics processes, each characterized by its own operational constraints and decision-making challenges. These challenges become even more significant in large, fully integrated terminals such as PSA GP, where gate, yard, berth, and rail subsystems coexist and interact continuously. In such environments, inefficiencies or disruptions in one subsystem can quickly propagate to other parts of the terminal, affecting overall performance.

One of the most critical challenges in port terminal operations is congestion and capacity limitation, particularly within the gate subsystem. Gate congestion is primarily caused by the arrival of external trucks for container pickup and delivery. However, congestion is not restricted to the gate area alone and often extends to internal terminal zones where trucks circulate to access yard blocks and collect containers. This situation leads to increased truck turnaround times and reduced operational efficiency.

In recent years, sustainability considerations have further intensified this challenge. The growing number of trucks accessing port terminals contributes to higher emissions, making environmental impact an increasingly important issue in terminal operations. The absence of a Truck Appointment System (TAS), which should be noted is not due to a decision by PSA GP but rather to the lack of agreement within the trucking community, further complicates congestion management,

as truck arrivals cannot be effectively coordinated with yard resource availability. For this reason, a substantial body of research has focused on analytical, optimization, and simulation-based approaches to study gate congestion and evaluate alternative operational policies under realistic conditions.

Within the yard subsystem, one of the main challenges is the efficient allocation and coordination of handling resources and internal transport equipment. Inefficient resource allocation can result in unbalanced workloads, increased travel distances, and reduced system productivity. In addition, yard performance is strongly influenced by the storage and positioning of containers. Non-optimal container placement often leads to additional re-handling operations, commonly referred to as housekeeping movements, which increase operational costs and negatively affect terminal efficiency. Simulation approaches are frequently used to analyze yard dynamics and assess the impact of alternative storage strategies and resource deployment scenarios.

The berth subsystem also presents significant challenges due to uncertainty in vessel arrival times and potential delays. The coordination between vessel operations and the arrival of trucks and trains is essential to ensure smooth terminal functioning. Furthermore, container stowage planning on board vessels represents a complex decision problem that directly affects handling efficiency and vessel turnaround time. The assignment and coordination of quay cranes for vessel loading and discharging operations are additional challenges that are commonly addressed through optimization and simulation models.

The rail subsystem introduces further complexity to terminal operations. One of the main challenges concerns the evaluation of rail facility capacity in order to assess how effectively existing infrastructure is utilized and how its performance can be improved. In addition, the optimization of train loading and unloading processes, considering the interaction between trains, yard equipment, and internal transport vehicles, represents a critical operational problem. Simulation-based approaches are particularly suitable in this context, as they allow the realistic modeling of train movements, resource interactions, and operational constraints within the rail yard.

Within this context, this thesis addresses several of the challenges described above by combining data analysis, optimization methods, and simulation approaches. The objective is to support improved decision-making and enhance operational performance at the PSA GP terminal.

Chapter 2

Literature Review

As mentioned in the introduction of this thesis, the optimization of container terminal operations has been widely investigated in the literature due to its significant impact on operational efficiency, environmental performance, and service quality. Over the past decades, researchers have proposed a variety of analytical, simulation-based, and optimization-driven approaches to address different aspects of terminal operations. This section reviews the most relevant contributions related to the scope of this thesis, with particular emphasis on truck (gate) operations and rail yard processes. The objective is to summarize existing methodologies, highlight their strengths and limitations, and identify research gaps that motivate the proposed framework.

Given the focus of this thesis on landside terminal operations, the state of the art is structured around the two main subsystems investigated: the truck subsystem and the rail subsystem. The first part reviews the literature related to truck-based container flows, including prediction, emission assessment, and congestion mitigation. The second part focuses on rail terminal operations, addressing train handling, scheduling, and rail yard capacity analysis.

2.1 Truck subsystem in container terminals

Starting from one of the main research objectives, related to the prediction of export and import containers arriving at or leaving the terminal by truck, it can be observed that a substantial body of literature already addresses the forecasting of container and truck flows in port terminals. Existing studies investigate this problem using a wide range of methodologies and modeling approaches, often relying on different types of information depending on the specific terminal configuration, data availability, and operational context considered in each case study.

In the context examined in this thesis, the prediction problem is approached from the perspective of terminal operations, consistently with the other problems addressed in the research. As

a result, the selection of input data and the adopted modeling choices are closely aligned with the characteristics of the specific terminal under study. While the general prediction problem has been widely investigated in the literature, the diversity of terminal layouts, operational practices, and available information implies that prediction models are often case-dependent. Consequently, approaches developed for a specific terminal context may offer insights and methodological elements that are relevant for similar terminal configurations.

In recent decades, Artificial Intelligence (AI) and Machine Learning (ML) techniques have increasingly been applied to prediction problems in port and terminal operations. Among these techniques, Artificial Neural Networks (ANNs) have emerged as powerful tools for modeling complex, nonlinear relationships in logistics systems. Early applications include the use of a priori knowledge neural networks to analyze relationships between port logistics capacity and regional economic development [27], as well as fuzzy Group Method of Data Handling (GMDH) neural networks for container terminal demand forecasting [28].

Several studies focus on vessel-related uncertainty as a key factor influencing truck flows and terminal operations. In [29], ship delays are forecast using neural-network-based learning algorithms, demonstrating how improved delay prediction can support more effective resource allocation. The importance of predicting vessel delays for operational optimization and cost reduction is further highlighted in [30]. Complementary approaches include the use of k-nearest neighbors (kNN) algorithms to predict vessel arrivals at ports [31], confirming the relevance of data-driven methods for modeling ship-related processes.

More recent research has shifted attention toward operational-level prediction of container and truck flows. In [32], a Long Short-Term Memory Recurrent Neural Network (LSTM RNN) is employed to forecast daily container volumes entering the storage yard prior to vessel arrival, with specific emphasis on data preprocessing and outlier treatment. Similarly, [33] proposes a two-stage prediction framework combining Deep Belief Networks (DBN) for feature extraction and Support Vector Machines (SVM) for truck arrival forecasting, exploiting the complementary strengths of both techniques. Hybrid approaches are further explored in [34], where SVM models are combined with back-propagation neural networks to predict container traffic volumes.

Short-term truck traffic forecasting has also received attention in the literature. In [35], a feed-forward neural network is used to predict short-term variations in outbound truck volumes generated by major container terminals, highlighting the relevance of high-resolution forecasts for real-time traffic management. Additional contributions focus on predicting related operational indicators, such as container dwell time. In [36], ANNs are applied to identify key factors influencing dwell time, supporting improved yard planning and allocation of handling equipment and human resources, and addressing challenges discussed in [37, 38].

Several studies explicitly incorporate vessel-related information as an input for predicting truck arrivals associated with both import and export operations. In [39], a Backpropagation Neural Network (BPN) is used to forecast inbound and outbound truck volumes using vessel data,

emphasizing the importance of appropriate feature selection for improving predictive accuracy. More recently, deep learning approaches have been proposed to capture complex temporal dependencies and external influences. In [40], truck arrivals are categorized according to their operational purpose and predicted using a hybrid Gated Recurrent Unit (GRU) and Fully Connected Neural Network (FCNN) architecture, revealing the impact of factors such as weekdays, weather conditions, and yard closing times. Similar deep learning-based forecasting approaches are further explored in [41].

Beyond prediction accuracy, several studies highlight the operational implications of truck arrival forecasting for congestion management and terminal performance. High truck volumes can lead to congestion inside and outside terminal boundaries, increased accident risk, and reduced service levels. Simulation-based analyses demonstrate how unmanaged truck flows generate inefficiencies and external costs such as congestion and air pollution [42]. In response, Truck Appointment Systems (TAS) have been proposed as coordination mechanisms to regulate truck arrivals [43]. Empirical and simulation-based studies show that TAS adoption can reduce truck turnaround times, smooth arrival patterns, and improve yard and crane productivity [24, 25]. Accurate truck arrival prediction is widely recognized as a prerequisite for the effective implementation of such systems.

Overall, the literature demonstrates that prediction of truck-based container flows, covering both import and export operations, has evolved from strategic-level throughput forecasting toward short-term, operationally oriented models. ANN-based, hybrid, and deep learning approaches have proven effective in capturing the complex interactions between vessel schedules, terminal operations, and truck movements. However, while substantial progress has been made in improving predictive accuracy, many studies focus on prediction as an isolated task, with limited integration into downstream decision-support processes such as congestion mitigation, emission reduction, and coordinated operational optimization. For this reason, in the present thesis, prediction outputs are not treated as an end in themselves but are instead used as input to subsequent analyses aimed at addressing emission and congestion mitigation problems in container terminal operations.

Beyond the prediction of truck-based container flows, a substantial body of literature addresses the environmental impacts associated with landside terminal operations. Truck traffic related to container handling represents a significant source of local emissions, particularly under congested conditions characterized by queuing at terminal gates, idling, and low-speed movements within terminal areas. As a result, several studies extend the analysis of terminal performance beyond purely operational indicators by incorporating emission estimation models and environmental performance metrics, and by investigating operational strategies aimed at mitigating emission levels while maintaining acceptable service standards.

Several studies address emissions in port contexts by proposing frameworks and methodologies for monitoring and evaluating greenhouse gas emissions associated with port and shipping activities. A comprehensive review of methods and criteria used to monitor CO_2 emissions from

ports and ships is provided in [44]. Similarly, [45] investigates the challenge of assessing the effects of emission reduction strategies in container terminals and shipping operations in situations where detailed energy consumption data are not available, highlighting the importance of modeling approaches in supporting environmental assessment.

Within port areas, truck traffic associated with container handling contributes to local emissions. Road transportation has therefore been extensively studied as a source of pollution, and numerous approaches have been developed to estimate and predict emissions on road networks with the objective of reducing their environmental impact. In this context, the application of traffic control and operational strategies aimed at jointly reducing congestion and emissions has been explored beyond the port domain. For instance, [46] proposes a two-class local ramp metering controller for freeway networks, demonstrating that appropriate control actions can simultaneously reduce congestion levels and lower CO_2 emissions. A key contribution of this work is the formulation of an emission model that links traffic flow conditions to pollutant generation.

In port-specific applications, different approaches have been proposed to estimate emissions generated by terminal operations. In [47], emissions at the Port of Rotterdam are estimated based on energy consumption of handling equipment during transshipment processes, accounting for factors such as equipment type, deployment, energy-consumption patterns, and travel distances. A similar energy-based logic is adopted in [48], where carbon emissions are estimated by multiplying the energy required at different stages of the container handling process by the corresponding carbon emission density of the consumed energy. The considered stages include vessel operations as well as receiving and delivery activities.

When focusing specifically on truck-related emissions, the selection of an appropriate emission model represents a critical aspect. Emission models differ in their level of detail and in the traffic variables they consider, ranging from aggregate emission factors based on average speed to more detailed formulations that account for driving conditions, congestion levels, stop-and-go behavior, and queuing. A comparative review of ten emission models for estimating greenhouse gas emissions from port truck operations is presented in [49]. The authors identify MOVES and COPERT as the most suitable models for port-related applications, while also highlighting their limitations in representing specific operational conditions typical of port terminals, such as frequent idling, congestion, and gate queues.

The COPERT model has been applied in several port-related studies to evaluate emissions from truck traffic. In [50], COPERT is used to estimate emissions generated by drayage trucks operating at the Port of Incheon, with the objective of quantifying the contribution of port-related trucking to urban air pollution and assessing potential policy measures for emission reduction.

In [51], vessel-dependent time windows are introduced as a gate management strategy, demonstrating significant reductions in truck waiting times and CO_2 emissions from both trucks and Rubber Tyred Gantry (RTG) Cranes during idle periods.

Beyond operational measures, several studies emphasize that emission reduction strategies of-

ten involve trade-offs between environmental objectives and operational performance. As highlighted in [52], optimizing truck arrival patterns requires balancing the reduction of emissions against the number of shifted or rescheduled truck arrivals, which may affect service levels and stakeholder acceptance.

Additional approaches to emission estimation combine traffic activity indicators with emission factors. In [53], emissions, including CO_2 , are estimated by multiplying vehicle kilometers traveled by emission factors associated with different vehicle types and average speeds using the US EPA's MOBILE6.2 software. Similar principles underlie applications based on COPERT, where emissions are computed as a function of traffic volumes and vehicle characteristics. More recent studies also explore data-driven methods to enhance emission prediction, as in [54], where a deep learning approach is combined with an amended COPERT model to address data sparsity and spatial heterogeneity in regional vehicle emission estimation.

Overall, the literature highlights that truck-related emissions in port terminals represent a significant environmental concern, particularly in urban settings. A wide range of emission estimation models and mitigation strategies have been proposed, including energy-based approaches, traffic-based emission models, and operational measures such as TASs and arrival time management.

In this thesis, the emission analysis builds upon well-established emission modeling approaches widely adopted in the literature, specifically the COPERT framework. Consistently with the predictive perspective discussed earlier, emission estimation is coupled with forecast information on truck arrivals at the terminal. This integration makes it possible to evaluate emission levels in advance, before truck traffic effectively occurs within the terminal. Such a forward-looking assessment enables the investigation of proactive emission mitigation strategies. In particular, anticipated emission levels are used to support the redistribution of truck arrivals over time, with the objective of reducing emission peaks while preserving acceptable operational performance. In this way, emission modeling is not treated as a standalone assessment tool, but as an integral component of a decision-support framework aimed at mitigating the environmental impact of truck operations in container terminals.

In addition to its role in supporting emission-related analyses, the availability of predicted truck arrival patterns also enables the investigation of congestion-related issues within the terminal. Truck traffic is not uniformly distributed across terminal areas, and congestion often arises at specific locations where trucks arrive to pick up containers, such as yard blocks or transfer points. Anticipating truck arrivals in both time and space therefore provides valuable information for managing traffic intensity within these critical areas.

In this thesis, predicted truck arrivals are further exploited to support the development of optimization schemes aimed at redistributing truck flows across time intervals and terminal areas. The objective is to mitigate traffic congestion and reduce localized overcrowding, while maintaining acceptable service levels for truck operations. In this context, prediction-based optimization is used as a decision-support tool to improve the spatial and temporal distribution of truck

movements within the terminal, contributing to smoother operations.

Congestion in maritime terminals has been extensively investigated in the literature due to its direct impact on operational efficiency, service levels for hauliers, safety, and environmental performance. Existing contributions analyze congestion phenomena in different terminal subsystems and from different methodological perspectives, including analytical modeling, simulation, and optimization-based decision support. Representative studies addressing congestion issues in terminal environments include, among others, [55, 56, 57]. At a broader level, congestion in multi-service port settings is examined in [58], while [59] proposes a logistics network design perspective, suggesting the coordinated use of multiple transport modes to reduce congestion effects across the overall network.

A relevant research stream focuses specifically on landside congestion, particularly at terminal gates and in areas where trucks interact with yard operations. One widely discussed approach is the concept of Vessel Dependent Time Windows (VDTWs), introduced in [60], where truck arrivals are grouped and assigned dedicated time windows with the aim of smoothing arrival patterns and improving gate management. Complementary analyses consider the broader operational and administrative factors contributing to congestion. For example, [61] investigates landside congestion in the ports of Nagoya and Hakata, highlighting the influence of documentary processes and their interaction with the overall timing of terminal operations. Yard-side congestion has also been studied in relation to internal transport disruptions: [62] addresses yard congestion by modeling yard truck interruptions through a combination of probabilistic and physics-based approaches.

In addition to analytical and optimization-oriented studies, simulation has been widely adopted to evaluate congestion mitigation policies under realistic operating conditions. A relevant example is provided by [42], which assesses the effectiveness of alternative algorithms within a truck management system already implemented in an Italian port. Such contributions highlight the value of simulation-based experimentation for comparing operational policies and understanding congestion dynamics that are difficult to capture through purely analytical models.

Among the most frequently discussed practical solutions in the literature, TASs are widely recognized as a key mechanism for managing gate arrivals and reducing congestion in port terminals. A TAS typically regulates truck arrivals by introducing a booking procedure through which truck operators reserve time slots for gate access. While this reduces hauliers' autonomy in selecting arrival times, it can lead to lower and more predictable service times inside the terminal and contribute to improved coordination between terminal and trucking operations. Accordingly, TASs are often considered a means of improving operational efficiency while also reducing the environmental impacts associated with congestion and idling [63].

A significant body of research addresses the design and implementation of TASs and the associated appointment scheduling problem. For instance, [43] proposes a mathematical optimization model to assign appointments to gate opening time slots, with the objective of guaranteeing pre-

defined service levels for hauliers while minimizing internal terminal congestion. Extending this direction, [64] presents a combined data mining and optimization approach to support truck operations management under a TAS. More broadly, [65] provides a comprehensive review of the literature on external truck appointment scheduling, synthesizing key approaches and identifying promising directions for future research.

In line with the emission-related analysis, truck arrival predictions are also used in this thesis to anticipate congestion in specific terminal areas and to design redistribution strategies aimed at smoothing truck flows and reducing localized congestion.

2.2 Rail subsystem in container terminals

The rail subsystem plays a strategic role in container terminal operations by enabling high-capacity and environmentally efficient hinterland transportation. In contrast to truck-based operations, rail transport allows the movement of large container volumes with lower unit emissions and reduced congestion impacts on surrounding road networks. For this reason, the integration and efficient management of rail facilities within container terminals have attracted increasing attention in the scientific literature.

Over the past years, a wide range of analytical, optimization-based, and simulation approaches have been proposed to study and improve rail terminal performance. Existing research addresses topics such as train arrival and departure scheduling, discharge and loading processes, resource allocation, and rail yard capacity evaluation. The following section reviews the most relevant contributions related to the rail subsystem in container terminals, with particular attention to scheduling models, exact and heuristic solution approaches, and simulation-based capacity analysis.

Focusing on the train discharge process, the literature highlights its critical role within rail terminal operations due to its direct impact on train turnaround times and downstream yard activities. Many contributions focus on specific subsystems, such as yard cranes, quay cranes, or truck operations, as illustrated in [66], [43], and [67]. While these studies provide valuable insights into individual components of terminal operations, the strong interdependencies among handling resources have motivated research toward integrated scheduling formulations.

Integrated scheduling problems in port terminals have been investigated using a variety of methodologies, including exact optimization models, heuristic algorithms, and metaheuristic approaches. A representative contribution is presented in [68], where an integrated scheduling model for handling operations in a container terminal railway area is proposed. The authors formulate a mixed-integer programming (MIP) model aimed at minimizing the operation time of Rail-Mounted Gantry Cranes, the travel time of Prime Movers, and the operation time of yard cranes. To address the computational complexity of the problem, a multi-layer genetic algorithm is in-

troduced.

The complexity of coordinating multiple types of handling equipment, combined with time-sensitive container handling and potential operational conflicts, is further highlighted in [69]. In this study, an integrated scheduling approach is proposed to reduce inter-equipment interference and to balance gantry crane workloads, improving overall operational efficiency.

Uncertainty in rail terminal operations has also been explicitly addressed in the literature. In [70], an integrated model is developed to optimize container prestaging, discharge, and loading rates at seaport rail terminals under uncertain conditions. The problem is formulated as a stochastic dynamic programming model, with the objective of minimizing expected logistics costs associated with container transportation and storage between yards and trains.

Earlier work on integrated scheduling can be found in the context of automated container terminals. In [71], a MIP model is proposed to improve terminal productivity by integrating operations across different types of handling equipment. Similarly, [72] formulates a MIP model that jointly considers vehicle scheduling and container storage decisions with the aim of minimizing ship unloading time. More recently, [73] addresses an integrated scheduling problem involving dual-trolley quay cranes, conflict-free automated guided vehicles, and dual cantilever rail cranes during loading and unloading operations. Given the NP-hard nature of integrated scheduling problems in automated terminals, the authors adopt a reinforcement learning-based hyper-heuristic genetic algorithm to obtain high-quality solutions within reasonable computational times.

Given the computational complexity of integrated scheduling problems in rail terminal operations, exact optimization approaches often become impractical for large-scale and realistic instances. For this reason, heuristic and metaheuristic methods have been widely investigated in the literature as effective tools for generating high-quality solutions within acceptable computational times.

Among these methods, Tabu Search (TS) has received considerable attention due to its capability to escape local optima through the use of adaptive memory structures and a strategic exploration of the solution space [74]. Since its introduction, TS has been successfully applied to a wide range of port and terminal optimization problems, including berth allocation, crane scheduling, and yard operations [75, 76].

Given the large number of decision variables characterizing real-world rail scheduling problems, TS-based heuristic approaches have proven effective in producing high-quality solutions within acceptable computation times, even in large-scale settings with complex operational constraints [77]. Furthermore, its relevance for integrated rail-port decision-making is emphasized in studies that jointly optimize container flows within port terminals and rail freight operations toward the hinterland. These works show that integrated approaches can significantly reduce both internal transport costs and railway delays compared to isolated optimization strategies [78].

In addition to TS, several other metaheuristic approaches have been proposed in the literature to address complex optimization problems in port and rail terminal operations, including Genetic Algorithms (GA) [79, 80] and Simulated Annealing (SA) [81]. These methods have demonstrated strong performance in handling large-scale and complex problem settings.

Beyond scheduling and operational optimization, the evaluation of rail terminal capacity represents another key aspect in the analysis of rail-based container operations. Capacity assessment is essential to understand the limits of the existing infrastructure and to identify potential bottlenecks affecting train handling performance.

In this thesis, the rail terminal capacity problem is addressed through two complementary approaches. First, an analytical model is developed to provide a theoretical evaluation of the rail facility capacity based on simplified representations of the operational processes. Subsequently, a discrete-event simulation model is employed to capture the dynamic behavior of the rail yard and train movements in greater detail, allowing for a more realistic assessment of terminal capacity under operational conditions.

A key challenge in rail freight transportation is the determination of rail system capacity, commonly expressed as the maximum number of trains that can be accommodated and served within a given time period [82]. Accurate capacity assessment is particularly relevant at strategic and tactical decision-making levels, where terminal operators must evaluate whether existing infrastructure and resources are sufficient to satisfy current and future demand.

Several analytical approaches have been proposed in the literature to assess railway capacity in broader transportation contexts. For instance, [83] introduces multi-objective models and techniques to analyze the absolute capacity of railway networks, providing insights into capacity evaluation from a system-wide perspective. Although such studies are not specifically tailored to port rail yards, they offer valuable methodological foundations that can be adapted to terminal environments.

More specifically, analytical and model-based approaches for evaluating capacity in port terminal contexts have been explored in [84] and [85]. These contributions investigate different aspects of terminal capacity using analytical formulations, standardized performance indicators, and multi-criteria evaluation frameworks, often in combination with scenario analysis.

Queue-based modeling has also been employed as an analytical tool to represent rail operations in seaport terminals. In [86], a discrete-time, queue-based planning approach is proposed to model railway operations associated with container handling processes in port environments. Such formulations provide simplified yet informative representations of rail yard dynamics and allow the identification of structural constraints affecting system performance.

Furthermore, simulation-based methodologies are widely recognized as effective tools for analyzing the stochastic and dynamic behavior of rail terminal operations. In contrast to analytical models, which often rely on simplifying assumptions, simulation approaches enable the explicit

representation of operational variability, resource interactions, and complex process logic. For this reason, discrete-event simulation and microsimulation have been extensively applied to evaluate rail yard capacity, identify bottlenecks, and support decision-making in both port and inland terminal contexts.

One of the early contributions in this area is presented in [87], where rail yard operations are modeled as interconnected queueing networks within a simulation framework. This approach allows the evaluation of yard processing capabilities and the identification of structural and operational bottlenecks affecting system performance. Building on similar principles, [88] propose a flexible simulation-based platform for assessing rail service infrastructures, supporting the evaluation of alternative operational scenarios and infrastructure configurations. In [89], a parametric railway capacity evaluation tool is introduced, combining simulation outputs with multivariable regression analysis to estimate capacity under varying operational conditions.

Microsimulation techniques have also been employed to assess rail terminal capacity at a detailed operational level. In [90], microsimulation is applied to evaluate the capacity of a freight village in Turin, enabling the analysis of terminal bottlenecks and performance under different traffic demand scenarios. Although this study focuses on an inland intermodal terminal connecting rail and road transport, it highlights the effectiveness of simulation-based approaches for capturing system dynamics and assessing capacity in complex logistics environments.

Several studies specifically address simulation-based capacity evaluation in port-related rail terminals. In [91], discrete-event simulation is used to analyze rail operations within a container port, demonstrating how train scheduling policies and handling coordination influence overall terminal capacity. Similarly, [92] develop a discrete-event simulation framework to evaluate the operational performance of the Ladkrabang Inland Container Depot (LICD) in Thailand, showing how equipment utilization, infrastructure layout, and operational rules affect terminal efficiency and throughput capacity.

More recent contributions emphasize the role of simulation as a strategic decision-support tool, particularly in the context of infrastructure planning and expansion. In [93], a discrete-event simulation methodology is proposed to predict operational bottlenecks and evaluate rail capacity in support of port expansion projects. The study highlights the importance of simulation-based analysis in situations where historical data are limited or when future demand scenarios must be assessed prior to large-scale investments.

Simulation frameworks have also been developed for broader intermodal and combined rail–road terminal contexts. In [94], a simulation-based approach is presented for the analysis of intermodal freight terminals, supporting the evaluation of operational policies and infrastructure utilization. A comprehensive overview of simulation applications in ports and container terminals is provided in [95], where the authors review a wide range of simulation-based studies and underline their relevance for capacity assessment, performance evaluation, and operational planning.

Overall, the literature indicates that analytical capacity evaluation models play a valuable role in

identifying the fundamental limits of rail yard performance by abstracting from detailed operational dynamics and supporting high-level assessments of infrastructure adequacy. In parallel, simulation-based approaches are shown to be essential for capturing the stochastic behavior of rail operations and the complex interactions among infrastructure components and resources. When used in combination, analytical and simulation-based methods provide complementary perspectives, allowing theoretical capacity estimates to be validated and refined under realistic operating conditions and supporting more informed planning and decision-making.

Chapter 3

System Architecture

This chapter provides an overview of the main subsystems of the PSA Genova Pra' (PSA GP) terminal and describes how they interact to form a complex and interconnected system. It also outlines the key operational processes relevant to the research studies conducted in this thesis, with a focus on explaining how these specific processes function within the terminal.

3.1 Subsystems and their interactions at PSA Genova Pra'

As mentioned earlier, in the case of PSA GP, the terminal benefits from the presence of berth, yard, gate, and rail facilities within a single integrated infrastructure, making it a representative example of a container terminal operating as a true system of systems.

Fig. 3.1 shows a view of the PSA GP terminal with its main components: the quay, the yard, and the connection to the hinterland provided by the truck gate and the rail park.

Although each subsystem fulfills a specific operational role, all subsystems are tightly interconnected through the movement of containers and the exchange of operational information, and they share critical resources, most notably yard space and handling equipment. As a result, the overall performance of the terminal depends on the coordinated functioning of all subsystems rather than on the isolated efficiency of individual components.

Terminal operations are largely governed by the berth subsystem, where vessel arrivals and departures define the main operational milestones. Deviations from planned vessel schedules, such as delays in vessel arrival, can propagate across the terminal and affect all other subsystems. In the export cycle, a delayed vessel may lead to shifts in truck arrival patterns, as export containers are delivered earlier or later than planned, altering gate congestion profiles and increasing pressure on yard storage capacity. Similarly, rail-delivered export containers may need to remain

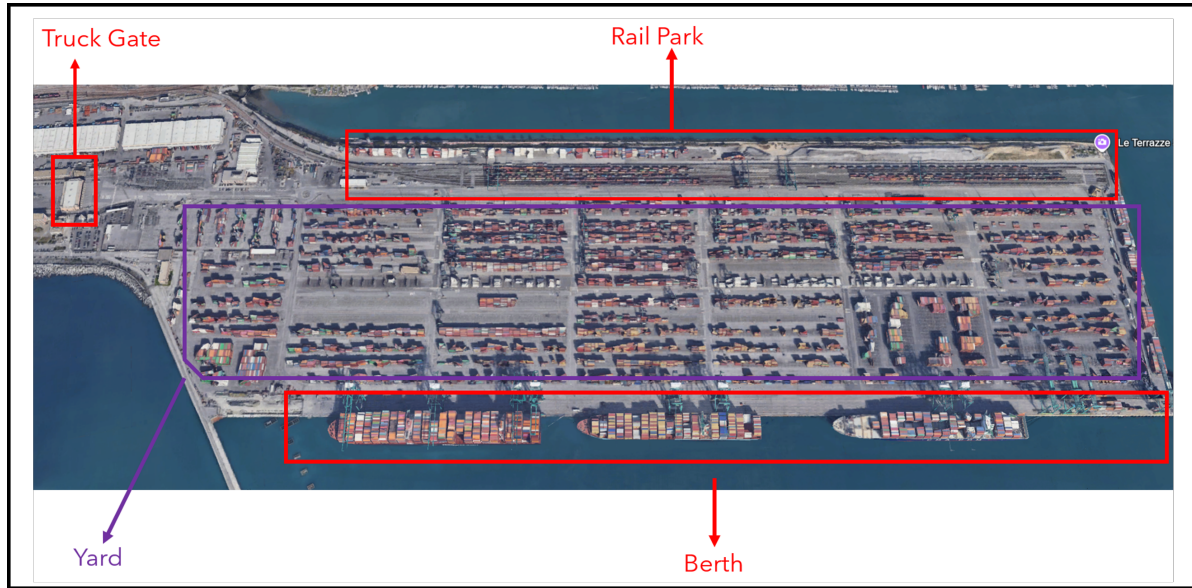


Figure 3.1: PSA GP main components (Source: Google Earth)

in the yard for longer periods, increasing yard occupancy and reducing operational flexibility. Elevated yard congestion can, in turn, slow down truck service rates and delay rail loading operations, potentially causing missed train departure slots.

Comparable interactions occur in the import cycle, where delays in container evacuation by trucks or trains can constrain yard capacity and interfere with ongoing vessel discharge operations. These examples illustrate how both physical container flows and information flows, such as arrival times, schedules, and resource availability, move continuously between subsystems and how local disruptions can escalate into system-wide effects, thereby motivating the system-level modeling and integrated scheduling approach adopted in this thesis.

In the following subsections, the operational processes considered and modeled in this thesis are described from a terminal and operational perspective, providing a detailed overview of how these processes are executed in practice. While the terminal comprises multiple interacting subsystems, the scope of this research focuses on selected processes that are most relevant to container flows and resource coordination. In particular, attention is given to gate and berth operations associated with container import and export via trucks, as these processes directly influence yard occupancy and handling dynamics. Furthermore, the rail subsystem is examined in greater detail, both in terms of facility capacity and operational constraints, and with specific emphasis on the optimization of train discharge operations. This focused analysis enables a detailed investigation of subsystem interactions while maintaining a system-level perspective on terminal performance.

3.1.1 Export process via truck in gate subsystem

Here a general view of the process of container exporting, starting from the containers arrival via truck, housekeeping process and finally the loading of containers onto the ship, is firstly given.

The whole process can be separated in three sub-processes:

- Truck Arrival to the Terminal for Container Discharge
- Housekeeping Process
- Container Loading onto the ship

Truck Arrival to the Terminal: Different stakeholders are included in the process of the truck arrivals to the terminal to bring containers for export.

It all begins with the customer which is in need of shipping a container.

To facilitate this, the customer contacts a freight forwarder, which is a company that takes care of the shipping and ensures the container is cleared for import or export. The forwarder, in turn, contacts the maritime agency, responsible for presenting a commercial offer. Following the customer's acceptance of the offer, the forwarder requests an empty container from the maritime agency and instructs the transport company to send a truck to the terminal to pick up the container. This container is then transported via truck to the customer's warehouse, where it is loaded with the customer's shipment. The process diagram of truck arrival in the terminal, developed using Bizagi Modeler, is presented in Fig. 3.2. Bizagi Modeler is a business process modeling tool based on the Business Process Model and Notation (BPMN) standard and is widely used to document, analyze, and communicate complex operational processes. It enables the structured representation of sequential activities, decision points, and interactions among resources, making it particularly suitable for modeling logistics and transportation processes. To avoid presenting a very large diagram, what was described above is included as a sub-process in the diagram.

Following the sub-process of documentation preparation for the delivery of the container to the terminal, a delivery order must be sent by the Maritime Agency to the forwarder before sending the truck to the terminal. When the order is released, the truck can transport the container to the terminal. Simultaneously, the Terminal Operating System (TOS) receives from Maritime Agency the container gate-in booking and all the data relating that container will be inserted in the TOS database. When the truck with the container arrives to the terminal, it initially stops at the so called One-Stop Office, which checks if there is a Gate-in booking for the container. In case of a booking existence the One-Stop Office performs the so called Administrative-Check (A-Check) of the container, which is then registered in TOS. Following the A-Check, the truck goes to terminal gate where the Physical Check (P-Check) of the container is performed. When passing the gate, the truck can directly go to discharge the container in the predefined position if

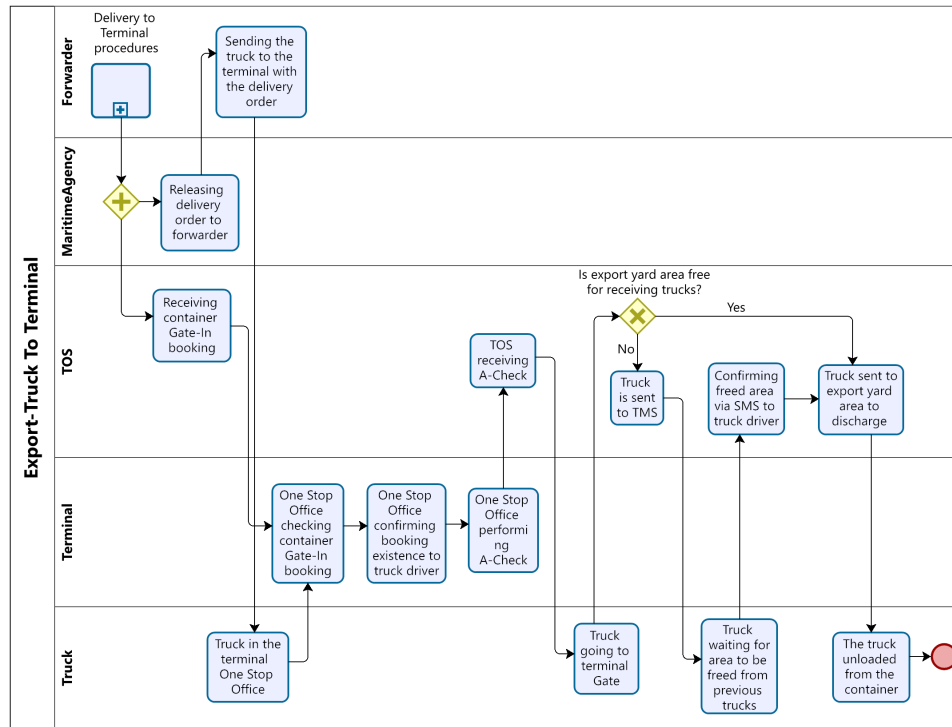


Figure 3.2: BPMN diagram illustrating the truck arrival in the terminal for container discharge

the specific area is free from trucks. Otherwise, in case the area where the container is destined to be positioned can not accept more trucks, the truck is positioned in a buffer zone. The truck will wait in the buffer zone until the area is cleared of previous trucks. Once the area is free, the driver will receive an SMS and/or an alert through a dedicated mobile application notifying them that they may proceed. Finally, the container is unloaded from the truck.

Housekeeping Process:

After being discharged from the trucks, the containers may not initially be placed in their optimal and final positions within the yard. Consequently, housekeeping or re-handling movements are required. These movements involve organizing the containers in stacks based on their weight class, length, type, and port of destination (POD). The grouping of containers in stacks is employed to ensure the smooth loading operations of the ship. Upon receiving container details from the TOS, the yard planners create a work-queue for container housekeeping. This work-queue is then sent to the dispatcher office, which, in turn, decides the teams responsible for the housekeeping process. The updated positions of the containers are recorded in the TOS. Subsequently, the yard planners may review the stack division to ensure accuracy and determine if additional housekeeping is necessary. As for the physical aspect of the housekeeping process, at PSA GP terminal in the export area the movement of containers is done using Reach Stackers

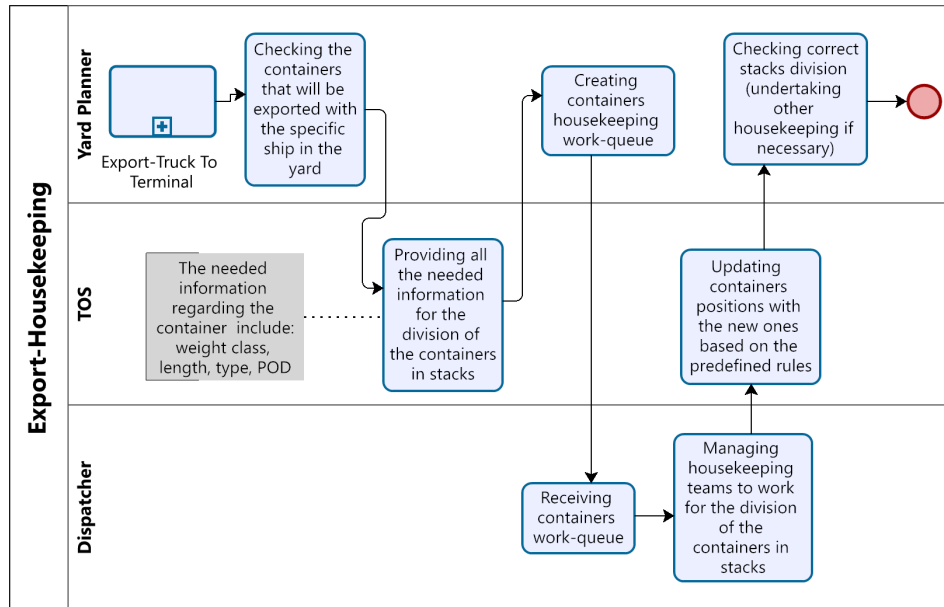


Figure 3.3: BPMN diagram illustrating the housekeeping process for exporting containers

(RS) and Prime Movers (PM). So, in case the container should be moved from position A to position B, the steps would be:

1. PM goes to position A
2. RS1 loads the container into PM
3. PM transport the container in the position B
4. RS2 discharge the container from the PM

Fig. 3.3 shows the process diagram of the housekeeping process.

Container Loading onto the ship:

Before a specific container is loaded into the ship, the custom forwarder must prepare the required export documentation for the container's loading. The container can only be loaded into the ship if the customs authority authorizes the documentation. In cases where the documentation is not authorized, the container cannot be loaded, and it should change the ship. This change will be registered in the TOS as a 'Change of Voyage' (COV) container. Regarding the physical process, which involves transportation within the yard and quay, the steps are as follows:

1. PM goes where the container is

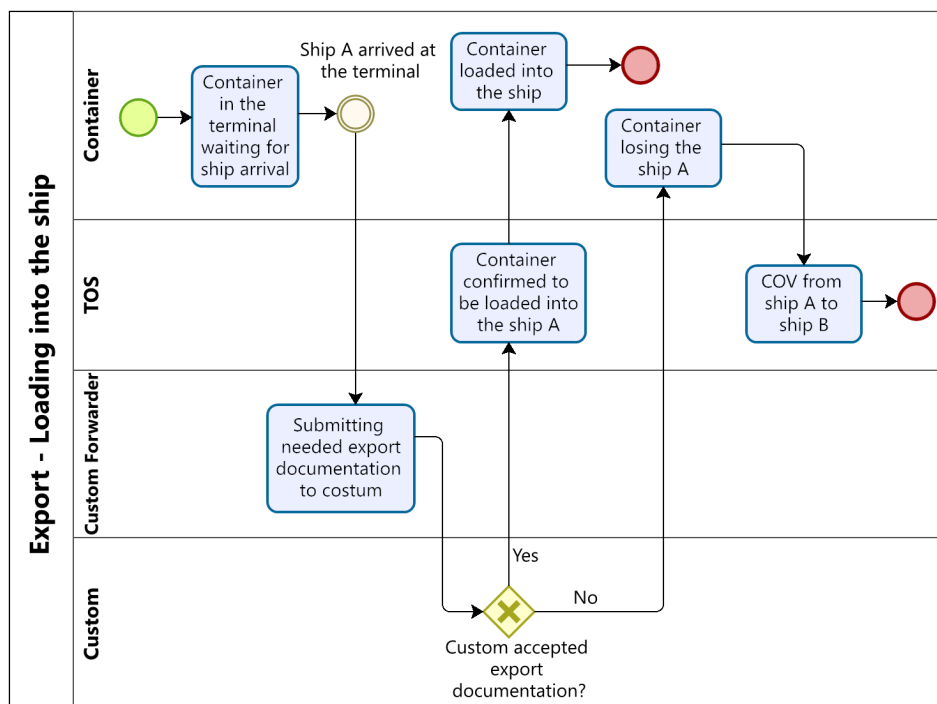


Figure 3.4: BPMN diagram illustrating the process of loading containers onto the ship

2. RS loads the container onto the PM
3. PM transport the container under the Quay Crane (QC)
4. QC discharge the container from PM and load it on the ship

The process diagram of container loaded into the ship is presented in Fig. 3.4.

3.1.2 Import process via truck in gate subsystem

The opposite process, namely the discharge of containers from the vessel followed by their transport by truck, is described here. The process of discharging containers from a vessel that are subsequently transported by truck generally follows the steps outlined below:

- the QC discharges the container from the ship and places it onto the PM;
- the discharged container undergoes a physical inspection, which may be performed automatically or manually in case of system failures;
- the PM transports the container to its assigned location in the yard;
- the rubber tyred gantry (RTG) crane unloads the container from the PM and stacks it in the designated yard block.

In most cases, trucks are allowed to enter the terminal for container collection only after a yard block has been fully populated with containers discharged from the same vessel. Before a truck arrives at the terminal, a set of administrative procedures involving the Forwarder, Customs Forwarder, and Maritime Agency must be completed, as also described in the export cycle. Once these procedures are finalized, the truck is authorized to access the terminal.

The container pick-up process generally follows the steps described below:

- the truck stops at the One-Stop Office to verify that all required documentation for container pick-up is in order;
- the truck passes through the terminal gate;
- the truck proceeds to the container pick-up area; if the area is congested, the truck may be required to wait in a buffer area until access is available;
- the RTG crane lifts the container from the stack and places it onto the truck;
- the truck exits the terminal.

This sequence applies when a truck enters the terminal solely for container collection. In the more common scenario where a truck both delivers a container and collects another during the same visit, the One-Stop Office must also verify the existence of a valid booking for the inbound container. The truck is first directed to the appropriate discharge area to unload the delivered container and is then routed to another area to complete the pick-up of the outbound container.

3.1.3 Train entry and exit in rail subsystem

As already mentioned, one of the main topics of this thesis is the evaluation of the capacity of the terminal's rail yard. Therefore, this section describes, from an operational point of view, the process of train entry and exit into the terminal and the activities it involves.

From a physical infrastructure perspective, the rail subsystem of PSA GP is composed of three main functional areas. The first area consists of six external tracks, commonly referred to as the Voltri Mare tracks, where arriving trains are initially received. These tracks serve as the interface between the national rail network and the terminal. One of the external tracks is not used for train arrivals or departures but is dedicated to locomotive bypass operations. The second area comprises two intermediate tracks, which act as a buffer and transition zone between the external tracks and the internal rail yard. The third area is the internal rail yard of the terminal, which is divided into two main parks, namely the East Park and the West Park, each containing multiple internal tracks used for loading and unloading operations.

The process of train entry and exit is represented through a detailed process diagram, as shown in Fig. 3.5.

When a train arrives at the external tracks, the external (line-haul) locomotive is decoupled from the train. A terminal maneuvering locomotive is then coupled to the train to enable internal shunting operations. The train subsequently moves through the intermediate tracks and enters the internal rail yard, where it is positioned on one of the designated internal tracks. Once the train reaches its assigned track, the maneuvering locomotive is decoupled, and container discharge and loading operations are performed using dedicated rail handling equipment.

After the completion of loading and unloading operations, the maneuvering locomotive is coupled again to the train, and the train undergoes a pressurization process. Train pressurization refers to the charging of the pneumatic braking system, during which compressed air is supplied through the main brake pipe to all wagons in the train. This process ensures that the brake system reaches the required pressure level, allowing the brakes on each wagon to be correctly released and guaranteeing safe and synchronized braking performance. Pressurization is a safety-critical operation and must be completed successfully before the train is allowed to move.

Once pressurization is completed, the train exits the internal rail yard via the intermediate tracks and reaches the external exit tracks of the terminal. At this stage, a series of technical and

operational checks are performed to verify correct container placement, load securing, and the proper functioning of onboard systems. The maneuvering locomotive is then decoupled, and the external locomotive is coupled to the train. A final brake test is conducted to confirm that the braking system responds correctly under the control of the external locomotive. Upon successful completion of these checks, the train is authorized to depart from the terminal and re-enter the national rail network.

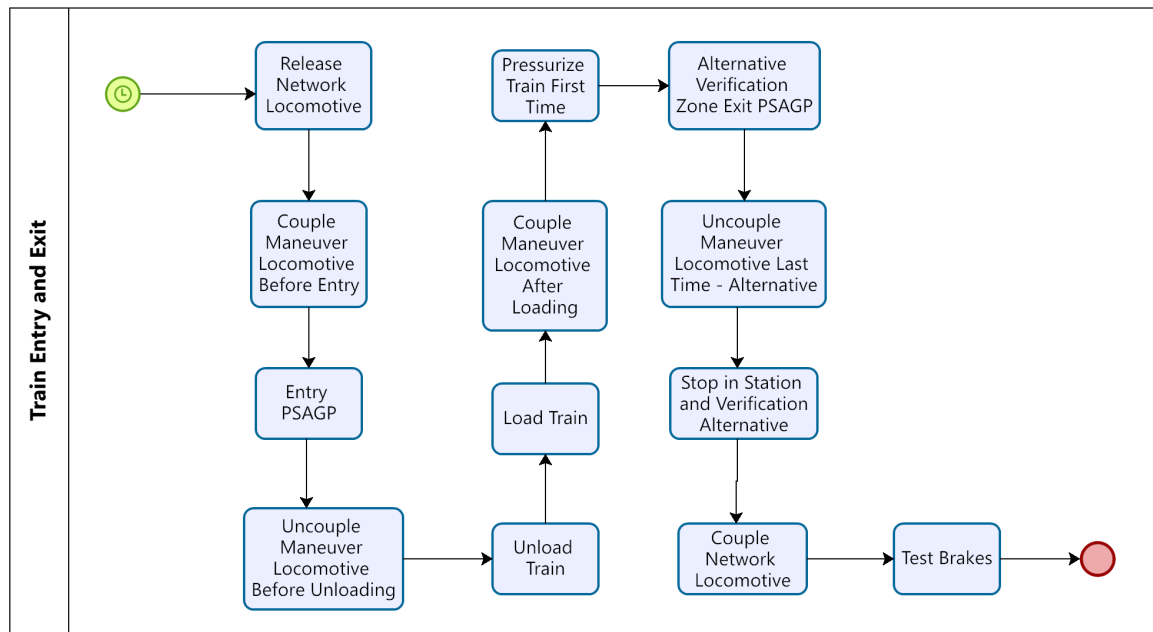


Figure 3.5: BPMN diagram illustrating train entry and exit in PSA GP

3.1.4 Train discharge in rail subsystem

The rail discharge process is described here in detail, as it represents one of the main processes analyzed and optimized in this thesis.

The process represented in the schema describes the first phase of rail container discharge operations, which covers all activities from discharge planning to the placement of the container onto PM.

The process starts with the Rail Planner, who initiates the operation by creating the Discharge

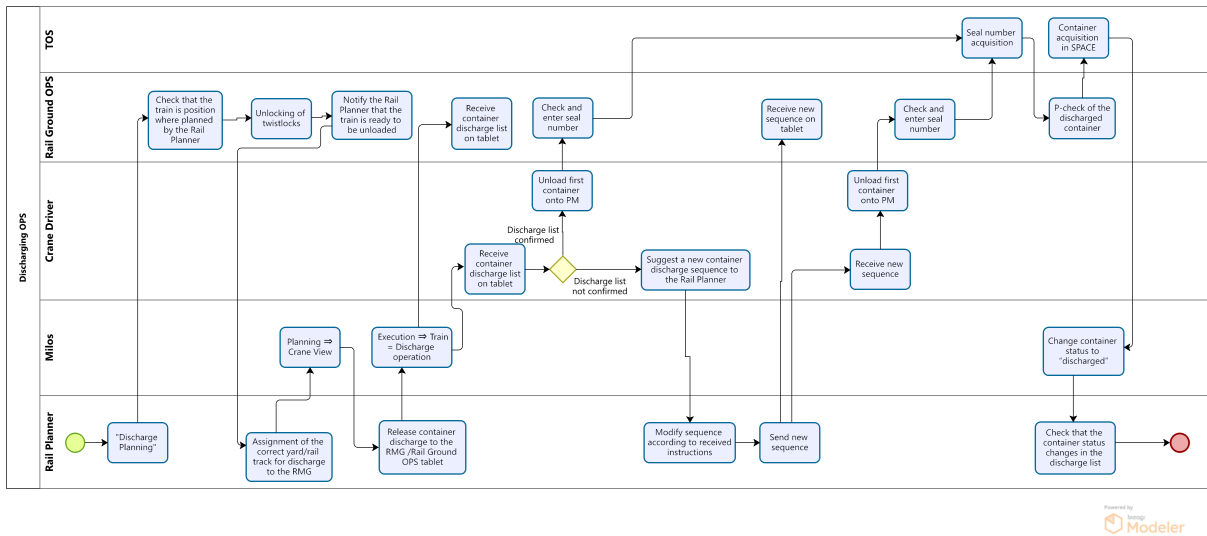


Figure 3.6: BPMN diagram illustrating the process flow of rail container discharge operations

Planning. Once the discharge plan is defined, control is transferred to Rail Ground Operations, which begin the operational validation phase.

The first operational step consists of Rail Ground OPS checking that the train is positioned where planned by the Rail Planner. Once the correct positioning is confirmed, the twistlocks are unlocked, enabling the physical discharge of containers from the train. Rail Ground OPS then notifies the Rail Planner that the train is ready to be unloaded, signaling that the execution phase can begin.

After Rail Ground OPS notifies the Rail Planner that the train is ready to be unloaded, control flows back to the Rail Planner, who performs the assignment of the correct yard and rail track for discharge to the Rail Mounted Gantry (RMG). Once this configuration is completed, the process moves to Milos, which functions as a rail TOS: it manages the registration of rail operations and supports the execution of rail processes. In Milos, the operation transitions from planning to execution by enabling the crane view and registering the train convoy as being in discharge operation mode.

Control then returns to the Rail Planner, who releases the container discharge tasks to the RMG and makes the discharge list available to Rail Ground OPS via their tablets. After the release, Rail Ground OPS receives the container discharge list on the tablet, and in parallel the Crane Operator also receives the container discharge list on the tablet, allowing operations to start.

At this point, the process reaches a decision step regarding the discharge list:

- If the discharge list is confirmed, the discharging continues according to the planned se-

quence;

- If the discharge list is not confirmed, the Crane Operator suggests a new container discharge sequence to the Rail Planner.

If a new sequence is required, control flows back to the Rail Planner, who modifies the sequence according to the received instructions and then sends the new sequence. The updated plan is distributed to the operational actors: Rail Ground OPS receives the new sequence on the tablet, and the Crane Operator receives the new sequence, after which unloading proceeds using the revised order (again starting with discharging the next container onto the PM).

After each container is placed onto the PM (whether under the original confirmed list or a revised sequence), control flows to Rail Ground OPS, which checks and enters the container seal number. The seal number is acquired by the central TOS, which performs seal number acquisition.

Following seal number acquisition, Rail Ground OPS performs a physical check (P-check) of the discharged container. Once the P-check is successfully completed, the container is registered in SPACE through container acquisition in SPACE. SPACE is the software system responsible for managing the positioning, tracking, and availability of containers within the terminal yard.

Then control flows to Milos, which updates the container status to “discharged” in the rail system. Finally, the Rail Planner verifies that the container status has correctly changed in the discharge list, ensuring consistency across planning, execution, and the involved IT systems.

This sequence repeats iteratively for all containers until the discharge list is completed. The process represented in the schema ends once containers are unloaded from the train and placed onto PMs, with correct status updates and registrations completed.

The process then continues with the PM transporting the container to the designated storage area. If the container is a full export container, it is transported to the export stacks, where a RS discharges it from the PM and stacks it accordingly. If the container is empty, it is transported to the dedicated empty container area, where it is again unloaded by a reach stacker and placed in the appropriate stack.

In the case of IMO containers, the unit is transported to the designated IMO area, where a RTG discharges the container from the PM and places it in the stack.

This sequence completes the train discharge process.

Chapter 4

Methodology

This chapter presents the theoretical background of the methods used to address the problems defined in this thesis and to achieve its objectives. In particular, it introduces the main methodological approaches adopted, including prediction models based on neural networks, optimization models, emission estimation models, and scheduling approaches based on both exact methods and heuristic and metaheuristic techniques.

For each of these components, the underlying principles and theoretical foundations are described, while the detailed model formulations, data, and specific equations are provided in the corresponding chapters.

4.1 Predictive models

An important component of this thesis is the development of predictive models for forecasting import and export container flows. These predictions support the improvement of terminal operations, which constitutes the primary focus of this research. By anticipating future container movements, predictive models can contribute to more efficient planning, resource allocation, and operational decision-making within container terminals.

Predictive models are computational techniques designed to estimate future or unknown outcomes based on previously observed data. They operate by learning relationships between input variables and target outputs and applying the learned patterns to generate predictions for new, unseen data. As described by [96], predictive modeling is a data-driven approach that utilizes historical and current data to forecast future outcomes and support decision-making processes. It integrates techniques from statistics, machine learning, and optimization to identify patterns and relationships within data and to produce reliable predictions.

In applied research, predictive modeling typically follows a structured process that includes problem definition, data collection, data preprocessing, model development, and model evaluation. The effectiveness of predictive models largely depends on the quality of the input data and the suitability of the selected modeling techniques. Following this systematic process helps ensure that predictive models are robust, reliable, and capable of generalizing to new data.

Within the context of logistics and transportation systems, predictive modeling plays a critical role in forecasting freight demand and vehicle flows. Transportation datasets often exhibit temporal dependency, seasonality, and nonlinear behavior, which makes predictive modeling techniques particularly suitable for such applications. Accurate forecasts of container and truck flows can support terminal operators in managing congestion, optimizing resource utilization, and improving overall operational efficiency.

Predictive models can be broadly categorized into regression-based models, time series forecasting models, and machine learning-based models. Time series approaches, such as the Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA) models, are commonly used to model temporal trends and seasonal patterns in transportation data. Regression models provide interpretable baseline predictions, while machine learning models offer enhanced capability to capture complex and nonlinear relationships among variables.

Among machine learning approaches, neural networks have gained increasing attention due to their ability to learn from historical data and generalize patterns in complex systems. These models are particularly well suited for transportation forecasting problems involving dynamic trade patterns and fluctuating container and truck flows.

According to [97], a neural network can be defined as an interconnected assembly of simple processing elements, referred to as units or nodes, whose functionality is conceptually inspired by biological neurons. The processing capability of the network is determined by the strengths of the connections between these units, known as weights, which are adapted through a learning process based on training data.

A typical neural network consists of an input layer, one or more hidden layers, and an output layer. Input data are propagated through the network by weighted connections, and each unit applies a nonlinear activation function to produce its output. By combining multiple layers of such units, neural networks are capable of modeling complex nonlinear relationships between inputs and outputs.

Learning in neural networks is commonly achieved through supervised training, where the network output is compared with known target values and the resulting error is used to update the connection weights. This process is repeated iteratively over a set of training patterns until the network converges to a configuration that minimizes prediction error. An important property of neural networks is their ability to generalize, meaning that once trained, they can produce accurate predictions for previously unseen data.

Feed-forward artificial neural networks trained using the backpropagation (BP) algorithm represent one of the most widely adopted architectures for supervised learning and prediction tasks. In feed-forward networks, information flows unidirectionally from the input layer through one or more hidden layers to the output layer, without forming feedback connections. The BP algorithm is commonly employed when target output values are known in advance, as it provides an efficient mechanism for minimizing prediction error in supervised neural networks [98].

The BP algorithm outlines a technique for computing the derivatives of the network training error with respect to the connection weights by applying the derivative chain rule [99]. The training procedure typically consists of two main phases: a forward propagation phase and a backward propagation phase [100]. During the forward phase, input data are propagated through the network layer by layer to produce an output. For each neuron, the weighted sum of its inputs is computed and processed by an activation function to generate the neuron output. This operation can be expressed mathematically as:

$$y_c = \varphi(z) = \varphi \left(\sum_{p=1}^n w_{cp} x_p + b_c \right) \quad (4.1)$$

where y_c denotes the output of neuron c , w_{cp} is the weight connecting neuron p in the previous layer to neuron c in the current layer, x_p represents the input to neuron c , b_c is the bias term associated with neuron c , and $\varphi(\cdot)$ is the activation function.

During the backward phase, the network weights are updated according to an error-correction rule. The error signal is obtained by comparing the network output with the desired target values, and this error is propagated backward through the network to adjust the weights in a direction that reduces the overall prediction error. The error function commonly adopted for regression problems is defined as:

$$E = \frac{1}{2} \sum_{c=1}^m (t_c - y_c)^2 \quad (4.2)$$

where m denotes the number of neurons in the output layer, t_c is the target output, and y_c is the actual output produced by the neural network. The corresponding weight update rule is given by:

$$\Delta w_{cp} = -\eta \frac{\partial E}{\partial w_{cp}} \quad (4.3)$$

where Δw_{cp} represents the update applied to the weight connecting neuron p to neuron c , η is the learning rate, and $\frac{\partial E}{\partial w_{cp}}$ is the partial derivative of the error with respect to the weight.

Various optimization strategies can be employed to improve the efficiency and convergence speed of the BP training process. Among these, the Levenberg–Marquardt (LM) algorithm is widely used due to its fast convergence characteristics. The LM algorithm combines features of the Gauss–Newton method and the steepest descent approach, providing a robust compromise between stability and computational efficiency [101].

To assess predictive performance, standard evaluation metrics such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) are commonly employed. These metrics provide quantitative measures of forecasting accuracy and enable objective comparison among different predictive modeling approaches.

4.1.1 Cross-validation strategy

Cross-validation serves as a systematic approach for evaluating the performance of models and making meaningful comparisons between different models [102]. Cross-validation operates by partitioning the available dataset into training and testing subsets, where the model is trained on the training data and subsequently evaluated based on its predictive performance on the test data [103].

Several variants of cross-validation exist, each defined by a different strategy for partitioning the available data. One of the most commonly used approaches is k -fold cross-validation, in which the dataset is divided into k approximately equal-sized subsets, referred to as folds. This procedure results in k distinct training–testing pairs, where in each iteration one fold is held out as the test set and the remaining folds are used for training. The k -fold scheme may be applied once using a single data partition, known as ordinary k -fold cross-validation, or repeated multiple times with different partitions, referred to as repeated k -fold cross-validation [104].

An important limiting case of k -fold cross-validation is leave-one-out cross-validation (LOOCV), which corresponds to the special case where the number of folds is equal to the number of available observations, i.e., $k = n$. In this approach, a single observation is excluded from the dataset and used as the validation sample, while the remaining $n - 1$ observations are employed for model training. This procedure is repeated iteratively until each observation has been used once as the validation sample. LOOCV offers the advantage of making efficient use of the available data, as nearly the entire dataset is used for training in each iteration. This property makes it particularly suitable for applications where the dataset size is limited or where preserving training information is critical. Moreover, by evaluating model performance across all individual observations, LOOCV provides a thorough assessment of the model’s generalization capability.

In this thesis, leave-one-out cross-validation is adopted to evaluate the predictive performance of the proposed neural network model. In addition to estimating overall generalization performance, LOOCV is also used as a diagnostic tool to analyze variations in prediction accuracy across different validation cases, thereby supporting a deeper understanding of the model’s behavior under different data configurations.

4.2 Emission estimation models

In the current global context, sustainability has become a central and increasingly complex concern, particularly in relation to port terminals, which can represent significant sources of environmental pollution. Given this context, the ability to anticipate operational impacts is of growing importance. Starting from a predictive model of the number of trucks accessing the terminal, the estimation of the associated emissions provides valuable insights into the potential environmental effects of terminal activities. Such an approach enables the identification of periods with increased emission levels and supports the adoption of proactive measures aimed at mitigating emission peaks and improving the overall environmental performance of terminal operations. Moreover, emission estimation models offer a practical alternative to direct emission measurements by providing systematic and scalable methods for quantifying emissions based on traffic activity and vehicle characteristics. In contrast, direct measurement through monitoring stations can be costly and spatially limited, particularly in complex transport environments.

Among the various emission sources associated with port operations, road freight transport represents a major contributor, especially due to the high volume of heavy-duty vehicles accessing terminals on a daily basis. Truck movements related to import and export activities can generate significant emission peaks, particularly during periods of high traffic demand.

One established approach for estimating emissions from road transport is based on the methodology presented in the 2023 edition of the COPERT Guidebook developed by [105] and published by the European Environment Agency (EEA). The guidebook provides a comprehensive framework for quantifying emissions associated with road vehicle activity across different vehicle categories, fuel types, and operating conditions. In this thesis, the COPERT methodology is applied to estimate emissions associated with truck movements related to import and export activities at a port terminal.

The emission estimation methodology covers a broad range of pollutants associated with road transport activity. These include regulated gaseous pollutants such as carbon monoxide (CO), nitrogen oxides (NO_x), non-methane volatile organic compounds (NMVOCs), methane (CH₄), and ammonia (NH₃), as well as greenhouse gases such as carbon dioxide (CO₂) and nitrous oxide (N₂O). In addition, sulphur oxides (SO_x), exhaust particulate matter (PM), and several toxic compounds related to fuel and lubricant composition, including polycyclic aromatic hydrocarbons (PAHs), persistent organic pollutants, and selected heavy metals, are also considered.

Within this framework, pollutants are classified into different groups according to the level of detail available and the estimation methodology applied.

Group 1 includes pollutants for which a detailed, activity-based estimation methodology is available. Emissions in this group are calculated using specific emission factors that account for vehicle technology, traffic situations, and operating conditions. The pollutants included in Group 1 are:

- Carbon monoxide (CO), expressed as CO;
- Nitrogen oxides (NO_x), expressed as NO₂ equivalent;
- Volatile organic compounds (VOCs), expressed as CH_{1.85} equivalent (also referred to as hydrocarbons, HC, in emission standards);
- Methane (CH₄);
- Non-methane volatile organic compounds (NMVOCs), expressed as VOCs (or HC) minus CH₄;
- Nitrous oxide (N₂O);
- Ammonia (NH₃);
- Exhaust particulate matter (PM), assumed to correspond to PM_{2.5};
- Particle number and particle surface area.

Group 2 consists of pollutants whose emissions are estimated primarily on the basis of fuel consumption. Although calculated using a different approach, the resulting emission estimates are considered to be of comparable quality to those obtained for Group 1. The pollutants included in Group 2 are:

- Carbon dioxide (CO₂);
- Sulphur dioxide (SO₂);
- Heavy metals contained in fuels and lubricants, including lead (Pb), arsenic (As), cadmium (Cd), chromium (Cr), copper (Cu), mercury (Hg), nickel (Ni), selenium (Se), and zinc (Zn).

Group 3 includes pollutants for which a simplified estimation methodology is applied, mainly due to limited availability of detailed emission data. This group comprises toxic compounds for which emissions are estimated using aggregated or speciated approaches, including:

- Polycyclic aromatic hydrocarbons (PAHs) and persistent organic pollutants (POPs);
- Polychlorinated dibenzo dioxins (PCDDs) and polychlorinated dibenzo furans (PCDFs);
- Polychlorinated biphenyls (PCBs) and hexachlorobenzene (HCB).

Group 4 consists of pollutants that are derived as fractions of total NMVOC emissions. In this group, certain compounds are estimated through NMVOC speciation procedures.

4.2.1 COPERT model

The COPERT methodology distinguishes between different levels of methodological detail for estimating emissions from road transport, commonly referred to as Tier 1, Tier 2, and Tier 3 approaches. These tiers reflect increasing data availability, modeling complexity, and estimation accuracy.

The Tier 1 approach represents the most aggregated level of emission estimation and is based on total fuel consumption data combined with fuel-specific emission factors. This method requires limited input data and is typically applied when detailed information on vehicle activity or fleet composition is not available. While suitable for high-level emission inventories, Tier 1 provides limited resolution and does not explicitly account for vehicle technology or operating conditions.

The Tier 2 approach introduces a higher level of detail by incorporating vehicle categories and emission-control technologies. Emissions are estimated using vehicle activity data expressed in terms of distance travelled, combined with technology-specific emission factors. Compared to Tier 1, this approach allows for a more refined representation of fleet composition and emission standards, but still relies on aggregated traffic information.

Tier 3 represents the highest level of methodological detail within the COPERT framework. Unlike Tier 1 and Tier 2 approaches, Tier 3 employs a vehicle-activity-based methodology that explicitly accounts for vehicle category, fuel type, emission standard, and traffic conditions. Emissions are calculated as a function of vehicle kilometres travelled and technology-specific emission factors defined for different driving environments, such as urban, rural, and highway conditions.

Implementation of the Tier 3 methodology requires detailed input data, including vehicle fleet composition, vehicle activity levels, average travelling speeds, and traffic conditions. By incorporating both technological and operational parameters, the Tier 3 approach provides a high-resolution and accurate estimation of emissions associated with road transport activity.

In this thesis, the Tier 3 methodology is adopted due to the availability of detailed information on truck activity and vehicle characteristics derived from the traffic prediction model. The use of a Tier 3 approach enables predicted truck flows to be directly translated into quantitative emission estimates, supporting the assessment of emissions associated with future import and export truck movements at the terminal. Furthermore, in the case study presented in a subsequent chapter in this thesis, only exhaust emissions are considered, while evaporative emissions are excluded from the analysis. This is justified by the focus on heavy-duty truck traffic, for which exhaust emissions constitute the primary source of pollutant emissions and are directly related to vehicle activity.

Under the Tier 3 methodology, total exhaust emissions are composed of hot emissions, generated when the engine operates at normal operating temperature, and cold emissions, which arise during transient engine operation before the engine reaches its optimal thermal state.

Then,

$$E_{total} = E_{hot} + E_{cold} \quad (4.4)$$

where:

- E_{total} are the total emissions (g) of any pollutant for the spatial and temporal resolution of the application;
- E_{hot} are the emissions (g) during stabilised (hot) engine operation;
- E_{cold} are the emissions (g) during transient thermal engine operation (cold start).

Hot exhaust emissions depend on factors such as vehicle travel distance, speed or road type, vehicle age, engine size, and vehicle weight. Since comprehensive data for all these parameters are often unavailable, emission estimation methods rely on the best available data. The calculation of hot emissions for a given time period is performed using experimentally derived emission factors, as described by the following expression:

$$E_{hot_{i,k,r}} = N_k \times M_{k,r} \times EF_{hot_{i,k,r}} \quad (4.5)$$

where, in the considered time period:

- $E_{hot_{i,k,r}}$ are the hot exhaust emissions of pollutant i [g], produced by vehicles of technology k driven on roads of type r ;
- N_k is the number of vehicles [veh] of technology k in operation;
- $M_{k,r}$ is the mileage per vehicle [km/veh] driven on roads of type r by vehicles of technology k ;
- $EF_{hot_{i,k,r}}$ is the emission factor in [g/km].

Vehicle technologies are defined according to vehicle category, engine type, and emission control standard. This classification is primarily based on the European emission standards (Euro classes), which distinguish vehicles according to their regulated emission performance. Vehicle categories include, among others, passenger cars, light commercial vehicles, heavy-duty vehicles, buses, and L-category vehicles. Each category is further subdivided by fuel type and Euro standard, resulting in a detailed representation of vehicle fleet composition.

The emission factor depends on the vehicle speed as reported in the following generic equation:

$$EF = \frac{\alpha v^2 + \beta v + \gamma + (\delta/v)}{\epsilon v^2 + \zeta v + \eta} \times (1 - RF) \quad (4.6)$$

where:

RF is a reduction factor, whereas $\alpha, \beta, \delta, \epsilon, \zeta, \eta$ are coefficients differing by fuel, vehicle class and engine technology.

As for the cold-start emissions, they are calculated in a similar way using the equation:

$$E_{cold_{i,k}} = \beta_{i,k} \times N_k \times M_k \times EF_{hot_{i,k}} \times \left(\left(\frac{EF^{cold}}{EF^{hot}} \right)_{i,k} - 1 \right) \quad (4.7)$$

where:

- $E_{cold_{i,k}}$ are the cold-start emissions of pollutant i , produced by vehicle of technology k ;
- $\beta_{i,k}$ is the fraction of mileage driven with a cold engine or the catalyst operated below the light-off temperature for pollutant i and vehicle of technology k ;
- N_k is the number of vehicles [veh] of technology k in circulation;
- M_k is the total mileage per vehicle [km/veh] in vehicle of technology k ;
- $EF_{hot_{i,k}}$ is the hot emission factor for pollutant i and vehicles of technology k ;
- $\left(\frac{EF^{cold}}{EF^{hot}} \right)_{i,k}$ is the cold/hot emission quotient for pollutant i and vehicles of technology k .

4.3 Optimization models

In logistics and transportation systems, optimization models are widely applied due to the complexity of operations and the presence of multiple interacting constraints. Such systems often involve limited resources, competing objectives, and dynamic demand patterns. Port terminals represent a critical component of logistics networks, where road, rail, and maritime transport interact within confined spatial and operational environments. In this context, optimization models play a key role in supporting decision-making by improving planning and operational efficiency, addressing challenges related to routing, scheduling, capacity allocation, and infrastructure utilization.

Optimization can be regarded as a systematic approach for supporting effective decision-making and prediction across a wide range of application domains, including production planning, engineering design, and financial analysis. In general terms, the optimization process begins with the formulation of a mathematical model that represents a specific decision problem, followed by the solution of this model using appropriate numerical algorithms. An optimization model typically requires the definition of a quantitative objective function, which represents a measure of performance to be maximized or minimized, such as benefit or cost. In addition, the model

incorporates a set of constraints that reflect physical, technical, or resource-related limitations on the feasible decisions. An optimal solution is defined as one that achieves the best possible value of the objective function while satisfying all imposed constraints [106].

From a solution perspective, optimization problems can be addressed using different classes of methods depending on their computational complexity and the guarantees required on the solution quality. As discussed in the optimization literature, these methods are commonly classified into exact methods and approximate methods. Exact methods aim to identify a globally optimal solution by exploring the feasible solution space, whereas approximate methods seek good-quality solutions with reduced computational effort, without guaranteeing optimality.

An algorithm designed for a minimization problem is referred to as an approximation algorithm if the worst-case solution it produces is guaranteed to be within a bounded factor of the optimal solution. This factor, greater than one, is known as the approximation guarantee and represents the maximum deviation of the algorithm's solution from the optimal value. In practice, however, it is often preferable to address large-scale optimization problems using heuristic approaches. Heuristics aim to identify suboptimal solutions efficiently, but their quality cannot be guaranteed analytically and is instead assessed through computational experiments. As a result, heuristic methods are typically employed for optimization problems that are not only difficult to solve exactly, but also challenging to approximate within guaranteed bounds [107].

Exact methods, and in particular mathematical programming techniques, play a central role in solving optimization problems when optimality, transparency, and reproducibility of the solution are essential. Despite the fact that many problems formulated as mathematical programming models are NP-hard, advances in modeling techniques and solver technology have made exact approaches applicable to a wide range of practical logistics and transportation problems.

4.3.1 Mathematical programming

Optimization models employed in logistics and transportation systems can take different forms depending on the nature of the decision problem and the characteristics of the underlying system. According to [108], mathematical programming encompasses several classes of optimization models, including linear programming, nonlinear programming, integer programming, and mixed-integer programming.

Linear programming

Linear programming is an optimization approach that represents a decision problem through a mathematical model. The term *linear* indicates that both the objective function and all constraints are expressed as linear functions of the decision variables. The term *programming* is used in the sense of planning rather than computer programming, referring to the systematic allocation of resources and activities in order to achieve an optimal outcome. Linear programming therefore

focuses on determining the best feasible solution, as defined by the mathematical model, among all possible alternatives. Although resource allocation problems constitute the most common application, linear programming is applicable to a wide range of decision problems. Any problem that can be represented using the general linear programming structure falls within this class.

A commonly used standard form of a linear programming model is:

$$\text{Maximize } Z = c_1x_1 + c_2x_2 + \cdots + c_nx_n, \quad (4.8)$$

subject to

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1, \quad (4.9)$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leq b_2, \quad (4.10)$$

$$\vdots \quad (4.11)$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \leq b_m, \quad (4.12)$$

and

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0. \quad (4.13)$$

In this formulation, the function being optimized is referred to as the *objective function*, while the restrictions on the decision variables are referred to as *constraints*. The first m inequalities represent the functional (or structural) constraints of the model, and the conditions $x_j \geq 0$ are known as nonnegativity constraints.

Nonlinear programming

Linear programming assumes that both the objective function and the constraint functions are linear. While this assumption holds for many practical decision problems, it is not always satisfied in real-world applications. When one or more relationships are nonlinear, the resulting optimization problem is formulated as a nonlinear programming model. In its general form, a nonlinear programming problem consists of determining a vector of variables $\mathbf{x} = (x_1, x_2, \dots, x_n)$ that optimizes a objective function subject to a set of constraints, when either the objective function or one or more constraints are nonlinear:

$$\text{Maximize } f(\mathbf{x}), \quad (4.14)$$

subject to

$$g_i(\mathbf{x}) \leq b_i, \quad \text{for } i = 1, 2, \dots, m, \quad (4.15)$$

and

$$\mathbf{x} \geq 0, \quad (4.16)$$

where $f(\mathbf{x})$ and $g_i(\mathbf{x})$ are given functions of the n decision variables. Nonlinear programming problems can vary significantly depending on the structure of the objective function and constraints. While certain classes of nonlinear problems can be solved efficiently, others may present substantial computational challenges, even for relatively small problem instances.

Integer Programming

One of the key assumptions underlying linear programming is the assumption of divisibility, which allows decision variables to take noninteger values. In many practical applications, however, this assumption is not appropriate, since decision variables often represent quantities that must be chosen in whole units, such as the assignment of vehicles, machines, or personnel to activities. When integrality requirements constitute the only deviation from the linear programming formulation, the resulting model is referred to as an integer programming problem.

From a modeling perspective, integer programming is obtained by extending the standard linear programming formulation by imposing integrality constraints on the decision variables.

A particularly important class of integer programming problems arises when decision variables represent yes-or-no choices. These decisions can be modeled using binary variables, which are restricted to take values in the set $\{0, 1\}$. In this case, a binary variable x_j can be defined as:

$$x_j = \begin{cases} 1, & \text{if decision } j \text{ is selected,} \\ 0, & \text{if decision } j \text{ is not selected.} \end{cases} \quad (4.17)$$

Integer programming models that involve only binary decision variables are commonly referred to as binary integer programming problems.

In this thesis, mathematical programming is used to formulate problems related to the minimization of congestion and emissions within the port terminal. It is also applied to model the problem of minimizing train discharge time and to determine the maximum number of trains that the rail yard can handle, which corresponds to the evaluation of its capacity.

4.4 Scheduling framework

In this thesis, scheduling models are employed to coordinate the timing and sequencing of train discharge operations in the rail yard part of the terminal, with the aim of efficiently allocating available rail and terminal resources over time.

Scheduling is a decision-making process that supports the allocation of limited resources to a set of tasks over specified time periods, with the objective of optimizing one or more performance criteria. It is a recurring activity in many manufacturing and service systems, where effective scheduling is essential for achieving efficient and reliable operations.

Both resources and tasks can take a wide variety of forms depending on the application context. Resources may include machines in a production facility, runways at an airport, crews at a construction site, or processing units in a computing environment. Similarly, tasks may correspond to operations in a manufacturing process, aircraft take-offs and landings, stages of a construction

project, or the execution of computational jobs. Each task may be characterized by attributes such as priority, earliest start time, or due date.

The objectives of scheduling problems also vary across applications. Common goals include the minimization of the completion time of the final task, the reduction of delays, or the minimization of the number of tasks completed after their respective due dates. One of the most commonly adopted objective functions in scheduling is the *makespan*, denoted by $C_{\max}$. The makespan is defined as the maximum completion time among all jobs, i.e.,

$$C_{\max} = \max\{C_1, C_2, \dots, C_n\},$$

where C_j represents the completion time of job j . Minimizing the makespan corresponds to reducing the time at which the last job finishes processing, and is often associated with improved utilization of the available machines and resources.

Over the past several decades, deterministic scheduling has attracted substantial research attention, leading to the development of a wide range of models and solution approaches.

In deterministic scheduling problems, the number of jobs and the number of machines are assumed to be finite. The number of jobs is denoted by n , while the number of machines is denoted by m . Throughout the scheduling framework, the subscript j is used to refer to a job, whereas the subscript i denotes a machine. If a job requires multiple processing steps, the pair (i, j) represents the processing operation of job j on machine i .

Each job j is characterized by a set of parameters that describe its processing requirements and scheduling constraints. The processing time of job j on machine i is denoted by p_{ij} . When the processing time does not depend on the machine, or when a job is processed on only one specific machine, the subscript i may be omitted and the processing time is denoted simply by p_j .

The release date of job j , denoted by r_j , represents the earliest time at which job j becomes available for processing in the system. The due date of job j , denoted by d_j , indicates the committed completion time of the job. Completion after the due date is typically permitted but may incur a penalty. When a due date must be strictly satisfied, it is referred to as a deadline and is denoted by $\bar{d}_j$.

In addition, each job may be assigned a weight w_j , which reflects its relative priority or importance within the system. This weight may represent, for example, the cost associated with keeping the job in the system, such as holding or inventory costs, or the value added by completing the job.

The machine environment specified in the α field defines the structural organization of resources within the scheduling system.

4.4.1 Scheduling environment classification

Several machine environments have been proposed in the literature to capture different operational settings [109].

Flow shop (F_m) refers to a system in which m machines are arranged in series and each job must be processed on all machines following the same fixed order. Specifically, jobs are processed first on machine 1, then on machine 2, and so forth until machine m . Upon completion on a machine, a job joins the queue of the next machine in the sequence. Queues are typically assumed to follow a First-In–First-Out (FIFO) discipline, implying that jobs cannot overtake one another while waiting.

The *flexible flow shop* (FF_c) extends the classical flow shop by introducing parallel processing capability at each stage. Instead of m machines in series, the system consists of c stages arranged sequentially, where each stage comprises a set of identical machines operating in parallel. All jobs must visit the stages in the same order, but at each stage a job requires processing on only one of the available machines. Each stage therefore functions as a bank of parallel resources. The queueing discipline between stages may or may not follow a First-Come–First-Served (FCFS) rule. Flexible flow shops are also referred to in the literature as hybrid flow shops or multi-processor flow shops.

In a *job shop* environment (J_m) with m machines, each job follows its own predefined processing route. Unlike the flow shop, different jobs may require processing on machines in different orders. A distinction is made between job shops in which each job visits each machine at most once and those in which a job may revisit the same machine multiple times.

The *flexible job shop* (FJ_c) generalizes the job shop environment by allowing multiple identical machines at each work center. Instead of m machines in series, the system consists of c work centers, each equipped with parallel machines. Jobs follow individual routes through the work centers, and at each visit a job requires processing on only one of the available machines.

Finally, the *open shop* (O_m) represents the most general machine environment. There are m machines, and each job must be processed on each machine exactly once, but no predefined order is imposed. The routing of each job is determined by the scheduler, and different jobs may follow different processing sequences. Some processing times may be zero, reflecting the absence of certain operations.

4.4.2 Resolution approaches

In many optimization problems arising in operations research, exact solution methods such as linear programming, integer programming, or nonlinear programming are capable of identifying optimal solutions. These approaches have proven highly effective for a wide range of practical

applications. However, as problem size and structural complexity increase, obtaining an optimal solution may become computationally infeasible. In such situations, it remains important to identify good feasible solutions that are reasonably close to the optimum.

4.4.2.1 Exact approach

Exact methods aim at finding optimal solutions to scheduling problems by exploring the solution space in a systematic and exhaustive manner. These approaches are typically based on mathematical programming formulations, such as Mixed-Integer Linear Programming (MILP), and solution techniques including Branch-and-Bound and Branch-and-Cut algorithms.

While exact methods are capable of guaranteeing optimality, their applicability is often limited to small- or medium-sized instances due to the exponential growth of the solution space as problem size increases.

In this thesis, the train discharge problem is first formulated as a MILP model and solved to optimality. However, the approach is computationally feasible only for small instances, as will be discussed in detail in the corresponding chapter.

4.4.2.2 Heuristics and metaheuristics

Heuristic methods are commonly employed to address optimization problems for which exact methods are impractical. A heuristic is a problem-solving procedure designed to efficiently search for high-quality feasible solutions without guaranteeing optimality. Although no formal assurance on solution quality can typically be provided, well-designed heuristics often produce solutions that are near-optimal or enable the identification of satisfactory solutions within acceptable computational time. Heuristic algorithms are frequently iterative in nature, progressively exploring the solution space and updating the best solution found during the search process.

Traditional heuristic methods are often based on relatively simple, problem-specific rules and are therefore tailored to particular applications. As a result, their applicability is generally limited to specific problem structures. To address this limitation, more general solution strategies known as metaheuristics have been developed. Metaheuristics provide a high-level framework and strategic guidelines for constructing effective heuristic algorithms that can be adapted to a broad class of optimization problems, rather than being restricted to a single application domain [108].

Certain planning and scheduling problems admit efficient solution methods and can be formulated as linear programs or solved using polynomial-time algorithms, allowing even large problem instances to be handled within reasonable computational effort. However, a substantial number of scheduling problems are inherently difficult and belong to the class of NP-hard problems. These problems cannot generally be solved to optimality using simple algorithms within limited

computation time, especially as the problem size grows.

Although some NP-hard scheduling problems can be formulated as integer or disjunctive programming models, solving such formulations to optimality may require excessive computational resources that are not always available in practice. Consequently, exact methods may become impractical for large-scale or time-sensitive applications. In these cases, heuristic approaches are commonly employed to obtain acceptable feasible solutions that are not guaranteed to be optimal but are sufficiently close to the optimum for practical decision-making [110].

Many heuristic algorithms used in scheduling belong to the class of local search methods. Unlike constructive heuristics, which build a schedule incrementally from scratch, local search methods start from an initial complete schedule and iteratively attempt to improve it. At each iteration, the current solution is modified according to predefined rules, and the resulting neighboring solutions are evaluated.

A local search procedure explores the neighborhood of the current schedule, where two schedules are considered neighbors if one can be obtained from the other through a well-defined modification, such as swapping or inserting jobs. Since local search does not guarantee convergence to a global optimum, its effectiveness depends strongly on the design of:

- the schedule representation;
- the neighborhood structure;
- the search strategy within the neighborhood;
- the acceptance or rejection criterion for candidate solutions.

Among the most widely used local search-based metaheuristics in scheduling are Simulated Annealing, Tabu Search, and Genetic Algorithms, which are briefly described below.

Simulated Annealing

Simulated annealing is a metaheuristic inspired by the physical annealing process in materials science, where a material is gradually cooled to reach a low-energy crystalline state. In the context of scheduling, simulated annealing performs a stochastic search over the solution space, allowing both improving and, with a certain probability, non-improving moves. Starting from an initial schedule, the algorithm selects a neighboring schedule and compares its objective function value to that of the current solution. If the new solution is better, it is accepted. If it is worse, it may still be accepted with a probability that depends on the difference in objective values and a control parameter often referred to as the temperature. This probability decreases over time according to a cooling schedule. The ability to accept worse solutions, especially in the early stages of the search, allows simulated annealing to escape local minima and explore a broader region of the solution space. As the temperature decreases, the algorithm becomes more

selective, gradually focusing on intensification around high-quality solutions. This balance between exploration and exploitation makes simulated annealing particularly effective for complex scheduling problems with rugged solution landscapes [110].

Tabu Search

Tabu search is a deterministic local search metaheuristic designed to prevent the search process from cycling back to previously visited solutions. Like simulated annealing, tabu search starts from an initial schedule and explores its neighborhood. However, instead of using probabilistic acceptance, tabu search relies on explicit memory structures. At each iteration, the algorithm selects the best admissible neighboring solution, even if it is worse than the current one. Certain moves or attributes of moves are temporarily declared tabu and stored in a tabu list, meaning they are forbidden for a specified number of iterations. This mechanism discourages short-term reversals and helps the search move away from local minima. An important feature of tabu search is the aspiration criterion, which allows a tabu move to be accepted if it leads to a solution that is better than any previously found. More advanced versions of tabu search incorporate additional memory structures and diversification strategies to further enhance exploration of the solution space. Tabu search has been successfully applied to a wide range of scheduling problems and is particularly well suited for problems where neighborhood evaluation can be performed efficiently [110].

Genetic Algorithms

Genetic algorithms are population-based metaheuristics inspired by the principles of natural evolution. In scheduling applications, each individual in the population represents a complete schedule, often encoded as a sequence of jobs or operations. The quality of each individual is measured through a fitness function, typically derived from the scheduling objective. The algorithm proceeds iteratively through generations. In each generation, individuals are selected based on their fitness to act as parents. New schedules, referred to as offspring, are generated through genetic operators such as crossover, which combines parts of two parent schedules, and mutation, which introduces random modifications. Unlike simulated annealing and tabu search, which operate on a single solution, genetic algorithms maintain and evolve a population of solutions. This population-based search enables a broader exploration of the solution space and reduces the risk of premature convergence to poor local optima. However, genetic algorithms often require careful tuning of parameters and may involve higher computational effort compared to single-solution heuristics [110].

In the context of port logistics and terminal operations, decision problems frequently involve large-scale systems, multiple interacting resources, and complex operational constraints. These characteristics can significantly increase computational complexity, making exact optimization approaches difficult to apply in certain cases. Under such conditions, heuristic and metaheuristic methods offer a practical alternative by enabling the identification of high-quality solutions within reasonable computational time, thereby supporting effective decision-making in complex

port and transportation systems.

The scheduling problem addressed in this thesis exhibit precisely these features, including large problem instances, tightly coupled rail and terminal resources, and operational constraints that arise in realistic port and rail yard environments. As a result, heuristic and metaheuristic methods provide a suitable and effective alternative, enabling the generation of high-quality feasible solutions within reasonable computational times, even when optimality cannot be guaranteed.

4.5 Queuing-based system representation

Queuing theory is introduced in this thesis as a conceptual framework for modeling capacity-constrained transportation systems and describing their state evolution.

Queuing theory is a branch of operations research that studies systems in which entities require service and may experience waiting due to limited service capacity. Such waiting phenomena arise in a wide range of practical contexts, including transportation systems, manufacturing processes, logistics operations, and communication networks. The presence of waiting has important implications for system efficiency, service quality, and operational costs, making queuing theory a valuable tool for analyzing and designing capacity-constrained systems.

At its core, queuing theory uses queuing models to represent systems in which entities arrive over time, join a queue if service is not immediately available, receive service, and then depart the system. These models provide a structured framework for describing how system components interact and how congestion and delays emerge as a result of limited resources. While queuing models are often employed to derive analytical performance measures, such as waiting times or system utilization, they may also be used at a conceptual level to represent the structure and dynamics of operational systems.

A generic queuing system consists of several fundamental components. Entities requiring service are generated by an input source, often referred to as the calling population. These entities enter the queuing system and wait in a queue until they are selected for service according to a specified queue discipline. The required service is then provided by a service mechanism, after which the entity exits the system. The state of the queuing system evolves over time as arrivals occur and services are completed.

The input source is characterized by the population from which arrivals originate and by the statistical pattern governing arrivals over time. Depending on the application, the calling population may be considered finite or infinite. In many modeling contexts, an infinite population assumption is adopted for simplicity, particularly when the number of potential entities is large relative to the system capacity. The arrival process is commonly described probabilistically, with assumptions regarding the distribution of interarrival times.

The queue itself represents the waiting space for entities that cannot be served immediately. Queues may be assumed to have infinite or finite capacity, depending on the physical or operational constraints of the system. The order in which entities are selected from the queue for service is determined by the queue discipline, with first-come–first- served being the most commonly assumed rule unless otherwise specified.

The service mechanism consists of one or more service facilities, each containing one or more service channels, referred to as servers. A queuing model must specify both the number of servers and their arrangement. In many elementary models, a single service facility with one or more parallel servers is assumed. The time required to serve an entity is referred to as the service time and is typically modeled using a probability distribution that reflects the characteristics of the service process [108].

Chapter 5

Simulation

An important part of the research work presented in this thesis is based on simulation. For this reason, this chapter is dedicated to the general concepts of simulation, as well as to the simulation software used and its underlying logic.

Simulation refers to the use of computational techniques to imitate the operation and behavior of real-world systems or processes over time. The system of interest may represent a physical facility, an operational process, or a complex interaction of resources and entities. In order to study such systems in a scientific manner, it is necessary to introduce a set of simplifying assumptions that describe how the system operates. These assumptions, typically expressed through mathematical or logical relationships, constitute a model of the system, which is then used to gain insight into system behavior and performance.

When the relationships governing a system are sufficiently simple, it may be possible to analyze the model using analytical methods and obtain exact solutions to the questions of interest. However, most real-world systems are characterized by high complexity, nonlinear interactions, and operational constraints that make analytical evaluation infeasible. In such cases, simulation provides a practical alternative by allowing the model to be evaluated numerically using a computer. Simulation outputs are then used to estimate key performance measures and assess system behavior under different conditions.

Simulation has been widely applied across a broad range of domains, including manufacturing systems, transportation networks, logistics operations, supply chains, and service systems. In the context of transportation and logistics, simulation has proven particularly valuable for analyzing complex infrastructures such as port terminals, rail yards, airports, and intermodal hubs, where multiple interacting processes and capacity constraints make analytical approaches impractical.

In studying a system, two general approaches can be distinguished: experimentation with the actual system and experimentation with a model of the system. Direct experimentation with the

real system may be desirable when it is feasible and cost-effective. However, in many practical situations such experimentation is either too expensive, disruptive, or impossible. In some cases, the system of interest may not even yet exist, and the objective is to evaluate alternative configurations or designs before implementation. For these reasons, it is often necessary to construct a model as a surrogate for the actual system and to conduct experiments on the model rather than on the real system. When a model is used for decision-making purposes, an essential issue is whether it accurately represents the system behavior relevant to the objectives of the study. This concern, commonly referred to as model validity, is a fundamental aspect of simulation methodology.

Moreover, experimentation with a model of the system may be conducted using either a physical model or a mathematical model of the system.

Physical models are tangible representations of a system, whereas mathematical models describe system behavior through quantitative and logical relationships. The vast majority of models used for operational analysis are mathematical models, as they allow systematic modification of inputs and evaluation of system responses. Once a mathematical model has been constructed, it must be assessed to determine whether it can be solved analytically or whether simulation is required. If an analytical solution to a mathematical model exists and can be obtained efficiently, it is generally preferable to use analytical methods. However, for complex systems, analytical solutions may be unavailable or computationally prohibitive. In such cases, simulation is employed to evaluate system performance numerically by observing how the model responds to different inputs and scenarios.

Simulation models may be classified along several dimensions. A fundamental distinction is made between static and dynamic simulation models. Static simulation models represent a system at a particular point in time, while dynamic simulation models describe the evolution of a system over time. Since many logistics and transportation systems evolve through sequences of events, dynamic simulation is commonly adopted in these contexts.

Another important classification distinguishes between deterministic and stochastic simulation models. Deterministic simulation models do not incorporate random elements, and their outputs are fully determined once the input values are specified. In contrast, stochastic simulation models include one or more random components, resulting in outputs that are themselves random and must be interpreted statistically. Many transportation, queuing, and logistics systems are inherently stochastic due to variability in arrivals, processing times, and operational conditions.

Simulation models may also be categorized as continuous or discrete. In continuous simulation models, system state variables change continuously over time, whereas in discrete simulation models state changes occur at distinct points in time, typically triggered by events. Discrete-event simulation (DES) is particularly well suited for modeling logistics and terminal operations, where system dynamics are driven by arrivals, departures, and the completion of processing activities.

Due to the complexity of modern operational systems, simulation has become an essential tool in

operations research and management science. When properly designed and applied, simulation models enable the evaluation of alternative system configurations, the analysis of operational policies, and the assessment of system performance under varying conditions [111].

5.1 Simulation for decision making

In today's economic and operational environment, decision-making is increasingly challenging due to uncertainty, variability, and the growing complexity of systems. Organizations operate in highly interconnected and dynamic contexts, where actions taken in one part of a system may lead to delayed, indirect, or non-intuitive consequences in another. As competition intensifies and resources become more constrained, decision-makers are required to evaluate trade-offs related to cost, service quality, risk, and innovation, often under time pressure and with incomplete information.

Many operational systems, such as manufacturing plants, logistics networks, service systems, and transportation hubs, are influenced by random events, time-dependent behavior, and complex interactions among system components. Examples include fluctuating customer demand, variable arrival times, machine breakdowns, shared resources, and human decision delays. These characteristics make it difficult to accurately predict system performance using simple analytical calculations or static planning methods. As a result, traditional approaches to decision-making are often insufficient when applied to modern operations systems.

To address these challenges, decision-support methodologies have been developed to assist decision-makers in understanding system behavior and evaluating alternative courses of action. Among these methodologies, simulation plays a central role. Simulation can be defined as experimentation with a simplified representation of a real system over time, with the objective of improving understanding and supporting system analysis and improvement. By replicating the structure, logic, and dynamics of a system in a virtual environment, simulation enables controlled experimentation without interfering with real operations.

Simulation-based decision-making allows decision-makers to analyze system performance, explore what-if scenarios, and assess the potential impact of proposed changes before implementation. Through simulation, alternative system designs, operational policies, and resource configurations can be tested and compared using quantitative performance indicators such as throughput, waiting times, utilization levels, service reliability, and cost. This significantly expands the decision space by allowing many alternatives to be evaluated efficiently and safely.

An important advantage of simulation is its ability to capture uncertainty and variability explicitly. While simulation may not predict exact future demand or events, it enables decision-makers to assess how variability affects system performance and to evaluate system robustness under different scenarios. In this way, simulation supports risk analysis by revealing possible outcomes

and identifying conditions under which system performance may degrade.

Simulation models are most effective when embedded within a broader decision-support framework that facilitates data input, experimentation, and result interpretation. Within such a framework, the simulation model acts as an intermediary between the real system and the decision-maker. Information from the real system is used to parameterize and validate the model, while simulation results provide insights that guide decision-making. The actions chosen based on these insights may then be implemented in the real system, leading to changes that can be further evaluated through subsequent simulation studies [112].

Fig. 5.1 presents a conceptual framework for simulation-supported decision making. The simulation model acts as an intermediary between the real system and the decision-maker, enabling experimentation with alternative scenarios and providing quantitative insights to support operational decisions.

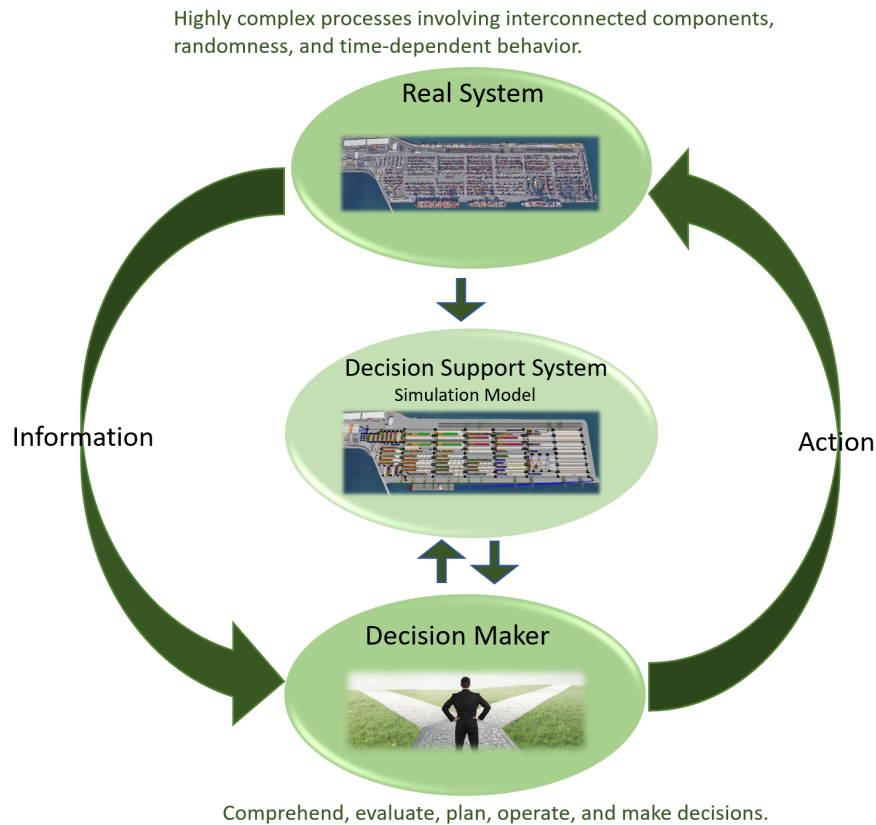


Figure 5.1: Decision-support framework based on simulation modeling

In the context of this thesis, simulation is employed not only as an analytical tool but primarily as a decision-support methodology. The objective is to support informed decision-making in

complex port terminal operations, where multiple interacting processes, shared resources, and stochastic elements make intuitive decision-making unreliable. By using simulation, alternative operational strategies can be evaluated systematically, supporting decisions that improve system performance while reducing risk and uncertainty.

5.2 Discrete-event simulation

Discrete-event simulation is a modeling approach used to represent the evolution of a system over time when changes in the system state occur at distinct, identifiable points in time. These points correspond to the occurrence of events, where an event is defined as an instantaneous action that may alter the state of the system. Between two consecutive events, the system remains unchanged. In more formal terms, the system state changes only at a countable number of time instants.

Although discrete-event simulation can theoretically be carried out using manual calculations for very small systems, real-world applications typically involve large amounts of data and complex interactions. For this reason, discrete-event simulation models are almost always implemented and executed on digital computers.

Discrete-event simulation is particularly suitable for modeling complex operational systems, such as transportation terminals, rail yards, and port facilities, where entities compete for limited resources and system dynamics are driven by arrivals, departures, and service completions.

Time-Advance Mechanisms: Because discrete-event simulation models describe dynamic systems, it is necessary to keep track of simulated time as the simulation progresses. This is achieved through a simulation clock, which stores the current value of simulated time. The units of simulated time are defined implicitly by the input parameters of the model and are independent of the actual computer execution time.

Two main approaches have been proposed for advancing simulated time:

- Fixed-increment time advance, where time progresses in equal steps.
- Next-event time advance, where the simulation clock jumps directly to the time of the next scheduled event.

In practice, the next-event time-advance mechanism is the most widely used, as it avoids unnecessary computation during periods where no state changes occur. Under this approach, simulated time advances from one event occurrence to the next, skipping intervals of inactivity. In the next-event time-advance approach, the simulation clock is initialized to zero, and the times at which future events will occur are determined. The simulation clock is then advanced to the time of the

most imminent event. At that moment, the system state is updated to reflect the occurrence of that event, and information about future events is revised accordingly.

This process is repeated iteratively: after each event, the simulation clock advances to the next scheduled event, the system state is updated, and new events may be generated. The simulation continues until a predefined stopping condition is satisfied, such as reaching a specified simulation time or completing a given number of events.

Because state changes occur only at event times, the simulation efficiently bypasses idle periods, making this approach computationally effective for large-scale systems [111].

Simulation Flow: The simulation begins at time zero with the execution of the initialization routine. Control is then transferred to the main program, which repeatedly invokes the timing routine to identify the next event and advance the simulation clock. Once the next event is selected, the corresponding event routine is executed to update the system state and generate any future events.

After each event, the simulation checks whether the stopping condition has been met. If not, the process repeats. When the stopping condition is satisfied, the report generator is invoked to compute and present the desired performance measures, and the simulation terminates.

This structure is commonly referred to as the event-scheduling approach to simulation modeling. An alternative approach, known as the process-interaction approach, focuses on modeling the lifecycle of individual entities as they flow through the system. While conceptually different, process-oriented simulations are typically executed internally using the same event-scheduling logic [111].

5.3 Simulation software environment

One of the main research projects presented in this thesis is developed using the software tools FlexSim and FlexTerm, both provided by FlexSim. While FlexSim provides the core simulation environment and modeling components, the simulation models are implemented and executed in FlexTerm. This section presents the software tools used in the study, their main components, and the logic governing their operation.

5.3.1 FlexSim

FlexSim is a powerful simulation and analysis software that supports engineers, planners, and decision-makers in the design, evaluation, and operation of complex systems. The software enables the development of three-dimensional computer models that represent real-world systems, allowing their behavior to be analyzed over a compressed time horizon and at a significantly

lower cost compared to experimentation with the actual system.

As a what-if analysis tool, FlexSim provides quantitative performance indicators that support the comparison of alternative system configurations and operational strategies. Its realistic 3D animation capabilities, combined with detailed performance reports, allow users to identify bottlenecks, detect inefficiencies, and evaluate improvement scenarios within a relatively short time. By modeling a system prior to its physical implementation, or by testing operational policies before they are applied in practice, FlexSim helps reduce the risks typically associated with system start-up and operational changes. Consequently, improvements that would otherwise require long periods of trial-and-error experimentation can be achieved more efficiently through simulation.

From a technical perspective, FlexSim is classified as a discrete-event simulation software. It is designed to model systems whose state changes occur at discrete points in time as a result of specific events. Typical system states include conditions such as idle, busy, blocked, or down, while events may involve the arrival of entities, movement of items, processing activities, or resource failures. The entities processed within a discrete-event simulation model may represent physical products, customers, documents, tasks, or information flows.

These entities progress through a sequence of processing, queuing, and transportation activities, commonly referred to as a process flow. Each activity may require one or more resources, such as machines, conveyors, operators, vehicles, or tools. Resources can be stationary or mobile, dedicated to specific tasks, or shared among multiple activities, depending on the system being modeled.

FlexSim is a versatile tool that has been successfully applied across a wide range of industries, including manufacturing, logistics, aerospace, automotive, food processing, defense, financial services, and project management. In general, the problems addressed using FlexSim can be grouped into three main categories:

- service systems, focused on efficiently processing customers and their requests;
- manufacturing systems, aimed at producing the right products at the right time and cost;
- logistics systems, concerned with delivering products to the correct location within the required time frame and at minimal cost.

5.3.2 FlexSim environment terminology

The FlexSim simulation environment, which is also adopted by FlexTerm, is organized through a graphical user interface that integrates all tools required for model creation, configuration, execution, and analysis. The core of this interface is the model view, which displays the three-dimensional representation of the simulation model. Within this view, simulation objects are

positioned, connected, and arranged to represent the physical and logical structure of the system, allowing users to visually inspect layout, movement, and interactions.

In addition to the model view, the environment includes a tree view, which provides a hierarchical representation of the model structure and internal objects. This view allows users to navigate complex models, access object relationships, and manage model components that may not be easily visible in the three-dimensional layout.

Simulation objects constitute the fundamental elements of the 3D simulation model. Different object categories serve distinct roles within the system.

- Flow items represent entities that move through the model from one location to another, typically between fixed resources. Depending on the application, flow items may represent products, customers, documents, parts, or other items circulating within the system.
- Fixed resources are stationary objects that perform specific functions, such as generating entities, storing them temporarily, processing them, or removing them from the system. Typical examples include sources, queues, and sinks.
- Task executors are mobile objects capable of moving within the 3D environment to perform tasks such as transporting flow items or operating equipment. A common example of a task executor is the operator, which may represent human labor within the system.

The Library, located in the left pane of the interface, provides access to a wide range of predefined simulation objects that can be used to construct the model. Each object available in the library includes a set of configurable properties that determine its behavior and interaction with other components. The content of the library may vary depending on the modeling tools currently active; for instance, when specialized tools such as Process Flow are enabled, the library displays objects specific to those tools.

The left pane also includes the Toolbox, which is used to manage and activate additional modeling tools such as Process Flow, dashboards, and global data tables. These tools extend the modeling capabilities of the simulation environment and support advanced logic, data handling, and performance visualization.

The Properties pane, located on the right side of the interface, provides access to the configuration settings of the currently selected object or tool. Through this pane, users can define operational parameters, logic rules, and other attributes that control object behavior and system dynamics during simulation execution.

The simulation environment also includes dashboards, which are used to display performance indicators, charts, and statistical outputs during or after simulation execution. Dashboards support real-time monitoring and post-run analysis, enabling users to evaluate system performance and visually compare alternative scenarios.

The console is another component of the simulation environment and is primarily used to display messages, warnings, and debugging information generated during model execution. This tool assists in model verification and troubleshooting by providing feedback on simulation events and logic execution.

Simulation execution is controlled through integrated interface controls that allow users to start, pause, step through, and reset simulation runs. Additional interface elements support navigation within the model view, window management, and customization of the workspace layout. Together, these components form a unified simulation environment that supports both visual model development and systematic experimentation [113].

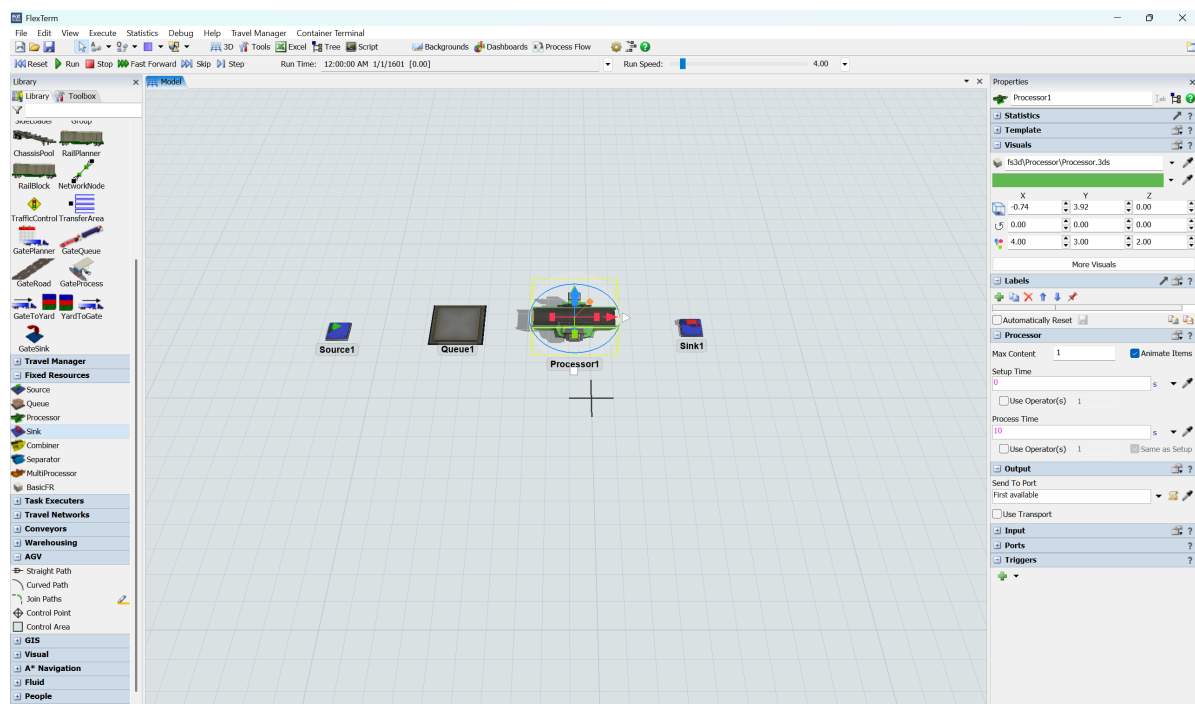


Figure 5.2: Simulation environment structure with selected fixed resources

Fig. 5.2 illustrates the simulation environment with main components and selected fixed resources, showing the basic flow of entities through generation, waiting, processing, and system exit stages.

5.3.3 Building model logic

In FlexSim (and similarly in FlexTerm), model logic refers to the overall behavioral rules that determine how 3D objects and other simulation elements interact while the model runs. In practice, model logic defines the sequence of actions and decisions that occur from the start of a

simulation run until its termination.

Model logic is typically built using a combination of configuration tools and, when needed, scripting:

- **Model parameters:** input variables used to control key aspects of the model (e.g., number of workers, arrival rates, rework probability, job sequences). They are especially useful for structuring scenario analysis because they can be systematically changed during experimentation.
- **Object properties:** the primary way to define and adjust the behavior and appearance of 3D objects through their settings (e.g., processing rules, routing logic, priorities, capacities).
- **Process Flow:** a central tool for building logic using a flowchart approach. The logic is constructed by connecting activities, each representing a logical step. During execution, tokens move through the activities and trigger the actions defined by those activities (e.g., assigning labels, delaying, moving flowitems). Process Flow can either (i) control the 3D model by linking tokens to 3D objects/flowitems, or (ii) implement self-contained logic inside the Process Flow itself.
- **FlexScript:** FlexSim's scripting language, used when logic requires deeper customization beyond configuration and Process Flow activities. It uses standard programming constructs such as commands, variable assignments, conditional statements, and dot syntax for accessing object properties and methods. FlexSim includes a code editor, which allows users to enter FlexScript or link to externally defined code. The code editor provides features such as syntax checking and code templates, helping users write correct and structured scripts for triggers, picklists, or custom object behavior. To support model development and ensure correctness, FlexSim also offers debugging tools. These tools allow the user to set breakpoints, step through code execution, inspect local and watch variables, and view the call stack.

Model behavior is driven by events that occur when objects interact with flowitems or when Process Flow activities process tokens. Objects also have states (e.g., idle, working, traveling, down) that are important for both operational logic and performance measurement. Logic can be attached through triggers, meaning that when a certain event or state change occurs, a defined action is executed (via object settings, Process Flow logic, and/or scripting).

A central mechanism for flexible logic in FlexSim is the use of labels, custom variables stored on tokens, flowitems, or 3D objects. Labels enable conditional routing, sorting rules, dynamic decision-making, and data tracking during the run. Conceptually, labels are used to store information (e.g., product type, destination, weight) and then reference that information inside object settings or Process Flow activities to change behavior. Labels can also store object references, which is a common way to link a Process Flow token to a specific 3D object or flowitem.

5.3.4 Modeling randomness

Simulation models frequently incorporate randomness to represent the inherent variability and uncertainty present in real-world systems, such as stochastic arrival times, processing durations, equipment failures, or routing decisions. In FlexSim, this randomness is typically introduced through probability distributions and random number generation, which allow the model to more realistically capture system behavior.

However, the use of randomness also affects model repeatability. To ensure that simulation results can be consistently reproduced and fairly compared across different scenarios, FlexSim provides mechanisms for controlling random number streams through the use of random seeds. By fixing the seed value, the same sequence of random numbers is generated in each simulation run, resulting in identical stochastic behavior when all other model conditions remain unchanged.

Controlling randomness is particularly important during model verification, debugging, and scenario comparison, as it ensures that observed differences in performance measures are caused by changes in model logic or input parameters rather than by random variation. Once model correctness has been established, multiple replications with different random seeds can be performed to capture statistical variability and obtain more robust performance estimates [113].

5.3.5 Simulation experiments and result analysis

To systematically evaluate alternative system configurations and operational strategies, simulation experiments in both FlexSim and FlexTerm are conducted using the Experimenter tool. The Experimenter provides a structured environment for defining, executing, and analyzing multiple simulation scenarios by varying selected model inputs and control parameters. This enables a consistent and repeatable approach to experimentation, which is essential for comparative analysis and decision support.

Model parameters considered in the experiments typically include inputs such as arrival rates, resource quantities, processing times, routing rules, and queue-related settings. For each experimental setup, the simulation model is executed according to the specified parameter values, and the relevant performance measures are automatically recorded.

To account for stochastic behavior and model randomness, each experiment can be executed over multiple replications. The Experimenter aggregates the output data across replications and provides summary statistics for each performance measure, such as average values and dispersion indicators. This supports statistically meaningful comparison between different parameter configurations.

The Experimenter generates structured output data in the form of tables and reports, which include the tested parameter values and the corresponding performance results. These outputs

allow the analyst to compare scenarios directly, identify trends, and evaluate trade-offs between alternative configurations. Performance indicators such as throughput, waiting times, utilization levels, and system delays are analyzed to assess overall system behavior.

The analysis of simulation results is based on comparing the aggregated performance measures across experiments in order to identify parameter settings that lead to improved system performance. Through this systematic experimentation and result analysis process, the Experimenter supports simulation-based decision making by providing quantitative evidence to guide operational and planning decisions in port terminal systems [113].

5.3.6 FlexTerm

FlexSim Container Terminals (FlexTerm) is a specialized simulation library developed within the FlexSim environment and specifically designed for modeling and analyzing container terminal operations. FlexTerm extends the general-purpose simulation capabilities of FlexSim by providing domain-specific objects, logic, and performance measures tailored to the operational characteristics of container terminals.

Simulation models developed using FlexTerm support the analysis and improvement of multiple aspects of terminal performance. In general, FlexTerm enables the evaluation of operational strategies aimed at increasing terminal throughput, improving equipment utilization, reducing waiting times and queue lengths, alleviating bottlenecks, and achieving a more balanced allocation of resources. In addition, FlexTerm models can be used to assess alternative investment options, evaluate new equipment or tool designs, and support training by illustrating overall system behavior and job-related performance.

More specifically, FlexTerm allows detailed analysis of key container terminal subsystems, including yard stacking and segregation strategies, yard layout and dispatching rules, gate operations and capacity, quay crane allocation, yard equipment assignment, hostler gang allocation, ship scheduling, berth assignment, and traffic constraints within the terminal. By modeling these interrelated processes in an integrated environment, FlexTerm supports the evaluation of complex interactions that are difficult to assess using analytical methods alone.

5.3.6.1 FlexTerm modeling paradigm

The FlexTerm modeling paradigm is based on the observation that container terminals worldwide share a common structural logic, centered on the handling and movement of containers through interconnected operational subsystems. These subsystems, as mentioned earlier in this thesis and reflected in the FlexTerm modeling logic as well, typically include the berth, where vessels arrive to load and unload containers; the gate, through which trucks enter and exit the terminal; the rail, where trains deliver and collect containers; and the yard, which serves as a temporary

storage and buffering area. Although these subsystems operate as distinct components, they are strongly interconnected, and container flow is governed by their interactions.

In FlexTerm, these subsystems are represented as interfaces, which define how containers enter, move through, and leave the terminal. The berth, gate, and rail interfaces may act either as sources or sinks of containers, while the yard functions primarily as a buffering system through which containers are routed between interfaces. In most cases, containers are temporarily stored in the yard for a specified dwell time before being transferred to their next destination.

A central concept in the FlexTerm modeling paradigm is the distinction between the Master Interface and Slave Interfaces. The Master Interface is responsible for generating container traffic within the model, while Slave Interfaces respond to this demand by providing or receiving containers as required. Containers are created in the model through operations defined at the Master Interface, whereas Slave Interfaces rely on supporting operations that accommodate the flow dictated by the master. The yard cannot act as a Master Interface, as it does not independently generate container traffic.

FlexTerm also makes extensive use of abstraction, allowing the modeler to focus on specific areas of interest while simplifying or omitting other parts of the terminal. For example, a model may represent only ship-side operations and the yard, while abstracting gate and rail activities, or it may focus solely on berth operations without explicitly modeling yard behavior. As long as at least one valid Master–Slave interface pair is defined, partial terminal models can be constructed efficiently. This abstraction capability enables targeted analysis while reducing unnecessary model complexity.

In this thesis, since one of the research projects focuses on the rail yard of the terminal, this subsystem is modeled in detail, while the remaining subsystems are represented in an abstracted form.

Each interface in FlexTerm represents not only a source, sink, or buffer, but also a complex operational system with its own equipment, schedules, and control logic. These elements are defined using planners, which provide structured interfaces for specifying arrival schedules, container operations, and equipment assignments. Container traffic is introduced into the model by defining operations within these planners, such as ship discharges, truck arrivals, or train movements. The correct definition of container origins, destinations, and supporting operations is essential to ensure logical consistency and avoid modeling errors.

Overall, the FlexTerm modeling paradigm provides a structured yet flexible framework for representing container terminal operations. By combining interface-based modeling, abstraction, and clearly defined container generation logic, FlexTerm enables the development of simulation models that are both analytically focused and operationally realistic [114].

5.3.6.2 Modeling rail yard in FlexTerm

The Rail Planner in FlexTerm is used to model rail-side operations of a container terminal, including train arrivals, container discharge and loading, track allocation, and work execution strategies. It provides a structured interface for defining the physical rail infrastructure, train compositions, schedules, and the logic governing rail operations, while offering default settings that allow rapid development of a basic rail model.

Rail modeling in FlexTerm follows a bottom-up approach. The lowest level of detail is defined through car profiles, which specify the composition of individual railcars in terms of container types, sizes, equipment classes, origins or destinations, and minimum and maximum quantities. These profiles determine how containers are generated or absorbed by the rail interface and play a critical role in defining whether the rail system acts as a data driver or responds to demand generated by another interface.

Based on car profiles, train profiles are defined by specifying the number and type of cars composing each train, along with arrival variability and loading or discharge configurations. Trains can include both inbound and outbound container flows, with minimum and maximum car counts allowing stochastic variation between train instances.

The Train Schedule defines when trains arrive at the terminal. Trains are placed on a weekly schedule using a graphical time-based view, allowing arrival times to be specified and adjusted visually. Arrival variability may be applied to represent uncertainty in real rail operations.

The physical rail infrastructure is represented through the Tracks configuration. Tracks are defined as ordered groups of rail blocks, determining where trains are positioned during operations. Track allocation logic specifies how arriving trains are assigned to available tracks, while abstract entry and exit times represent the delay between scheduled arrival and the start of work, as well as the time required for a train to clear the track after operations are completed.

Operational behavior during train servicing is governed by Work Strategies, which define how containers are discharged from or loaded onto trains. These strategies contain logic that coordinates equipment usage, container movements, and sequencing of operations while referencing the appropriate rail gang resources. The work strategy therefore represents the core operational logic of the rail interface.

Overall, the Rail Planner provides a comprehensive yet flexible framework for modeling rail yard operations within FlexTerm. By combining structured parameter definitions, scheduling, infrastructure modeling, and programmable work logic, it enables detailed analysis of rail-yard interactions while remaining consistent with the broader FlexTerm modeling paradigm.

5.3.6.3 Travel Manager

An important component of FlexTerm is the availability of the Travel Manager (TM) module. TM is a simulation package specifically designed to model horizontal transport systems in container terminal operations. It provides advanced routing and conflict management capabilities through a space-claiming mechanism, which is commonly used in automated horizontal transport systems. As a result, TM enables a more accurate simulation and analysis of transport operations within the terminal.

TM extends the standard FlexTerm functionality by incorporating vehicle movement resolution and collision avoidance. Compared to the default network node navigator, TM offers improved routing decisions and more realistic conflict management. This is particularly beneficial for modeling automated transport systems such as automated guided vehicles (AGVs) and automated straddle carriers. The main components of TM include the Multi-Lane Object (MLO), which represents the travel paths for horizontal transport vehicles (e.g., trucks, AGVs, and straddle carriers), and parking spots, which are designated locations where a traveler can stop. An important attribute is the *Parking Lane Only* setting: when a lane is defined as parking-lane-only, it is not used for regular travel but only when the destination parking spot lies on that lane and is sufficiently close [115].

The TM provides several key functionalities:

- Parking Categories are used to group parking spots according to their operational purpose, for example staging areas for vehicles before serving a specific resource. Multiple parking categories can be defined depending on the model requirements.
- Associated Parking Categories allow model objects to be linked to specific parking categories, enabling structured and context-dependent parking behavior.
- Travel Rules define how a parking spot is selected for a given movement, typically based on factors such as the current task, the destination, and whether the vehicle is loaded or empty.
- Travelers define the properties of moving entities (e.g., trucks), including kinematic parameters such as speed, acceleration, and deceleration.
- Finally, the General Settings include global configurations such as lane selection strategies and parking spot selection rules. In this thesis, the parking spot selection functionality is specifically used to determine where a prime mover should stop in order to perform loading or discharge operations with the rail-mounted gantry crane.

Chapter 6

Container Volume Prediction

Containers arriving at and departing from the PSA Genova Pra' (PSA GP) terminal by truck represent a significant share of the total container flows handled at the terminal. These truck-related movements are closely linked to vessel operations: containers arriving by truck are typically intended for export on a specific vessel, while containers leaving the terminal by truck are usually imported containers that have been previously discharged from a vessel. As a result, both the arrival and departure patterns of truck-handled containers are strongly correlated with vessel schedules.

Accurately predicting the number of containers arriving and leaving the terminal by truck is therefore highly valuable for yard planners, as it supports more efficient planning of yard stacking strategies and space allocation. Furthermore, container flow predictions can serve as a basis for forecasting truck arrivals. When such forecasts are available in advance, they can significantly contribute to preventing congestion at terminal gates and within operational areas, thereby improving overall terminal efficiency.

Below, a brief description of the external truck operations at the PSA GP terminal is provided in order to contextualize the subsequent analysis.

Trucking constitutes a critical component of container transportation at the terminal, accounting for approximately 78% of the total transport volume over the last three years. According to the 2023 data analysis, the PSA GP terminal handles on average about 2,100 trucks per day, with peak values occasionally exceeding 2,300 trucks per day, while roughly 500 trucks can be processed simultaneously. Such traffic levels pose a significant risk of uncontrolled truck flows, with daily arrivals potentially exceeding the terminal's processing capacity. This situation may lead to congestion in operational areas, increased accident risks, challenges in coordinating yard equipment operations, and growing frustration among truck drivers. Moreover, inadequate management of truck arrivals can result in long queues at terminal gates and excessive numbers of trucks circulating within the terminal, thereby reducing overall terminal efficiency and generating

substantial external costs, including traffic congestion, air pollution, and noise [42].

Concerning the truck arrival process, it should be noted that a formal Truck Appointment System (TAS) is not currently implemented at the PSA GP terminal, as its adoption is still under discussion with the local trucking community. A TAS, as previously mentioned, is a technological platform aimed at regulating truck flows at port terminals by supporting the scheduling and coordination of truck arrivals and related operations [43]. In the absence of such a system, trucks may arrive at the terminal in a largely unscheduled manner, subject only to a set of predefined operational constraints that will be described in detail later. This lack of structured scheduling further complicates the task of forecasting truck arrival patterns.

Nevertheless, truck arrivals are governed by specific temporal and operational restrictions. Trucks delivering or collecting containers are permitted to access the terminal every day except Sundays, within a time window from 6:00 AM to 10:00 PM. On Saturdays, this window is reduced, allowing access from 6:00 AM to 2:00 PM. In addition, trucks must comply with a predefined delivery interval, which begins a certain number of days prior to the scheduled arrival of the vessel. This delivery interval plays a crucial role in predicting container arrivals by truck and will be discussed in greater detail in the following sections. As described in the introduction chapter of the terminal, the gate facility is composed of twelve semi-automated lanes, evenly divided between inbound and outbound truck traffic, with six lanes allocated to truck entry and six to truck exit.

The first schematic in Fig. 6.1 summarizes the initial main part of this thesis, which addresses one of the primary research objectives related to the improvement of the truck subsystem. The framework is described at a high level, both from a software and a methodological perspective, in order to illustrate the logical flow between the different components of the system and to highlight how the output of each stage serves as the input for the subsequent one.

The process begins with the collection of relevant operational data from the company's data warehouse through structured SQL queries. Following data cleaning and preprocessing, the selected inputs are used to develop a prediction model implemented in MATLAB using neural networks.

The prediction model provides estimates of the number of containers imported and exported via trucks. Based on these estimates, the daily number of truck arrivals at the terminal is evaluated using appropriate approximations. Subsequently, a selected emission evaluation model is applied to estimate the emissions generated by truck traffic. On the basis of these results, an optimization scheme is developed to mitigate emission peaks on specific days by redistributing a portion of truck arrivals to days with lower emission levels.

The same methodological approach is also applied to mitigate truck traffic congestion within specific terminal areas. By leveraging the predicted truck arrivals, an optimization-based redistribution of trucks is performed, enabling a smoother temporal and spatial distribution of truck flows and contributing to improved operational performance of the truck subsystem.

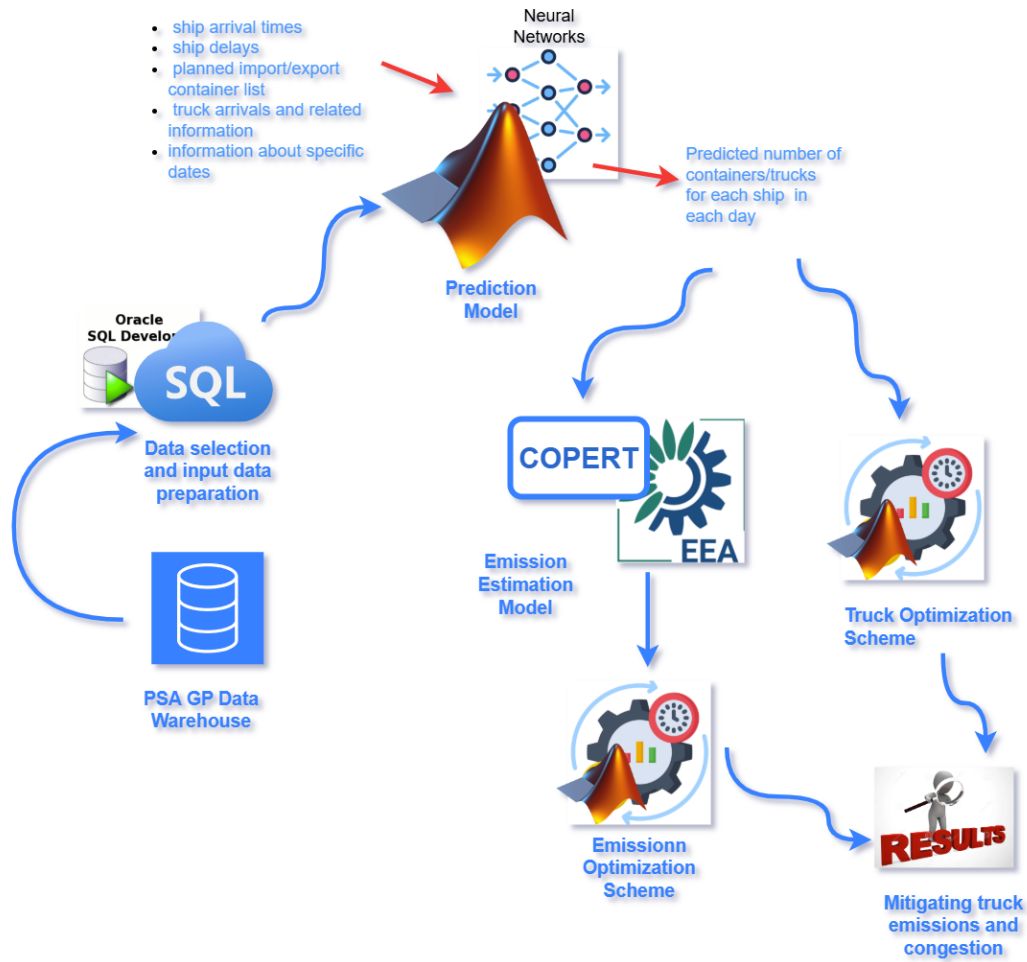


Figure 6.1: Conceptual framework of the research on the truck subsystem

6.1 Prediction of export containers via truck

This section focuses on the prediction of export containers transported by truck and presents the data and results associated with the proposed methodology. It starts with a brief overview of the process, followed by a detailed description of the predictive model and the dataset employed. The section concludes with a discussion of the obtained results.

6.1.1 Brief reminder of the export process

Before introducing the prediction model, a brief description of the container export process is provided. The process begins with containers arriving at the terminal by truck and being temporarily stored in the yard, as illustrated in Fig. 6.2. It then continues with the transport of containers from the yard to the berth, where they are loaded onto the vessel, as shown in Fig. 6.3.

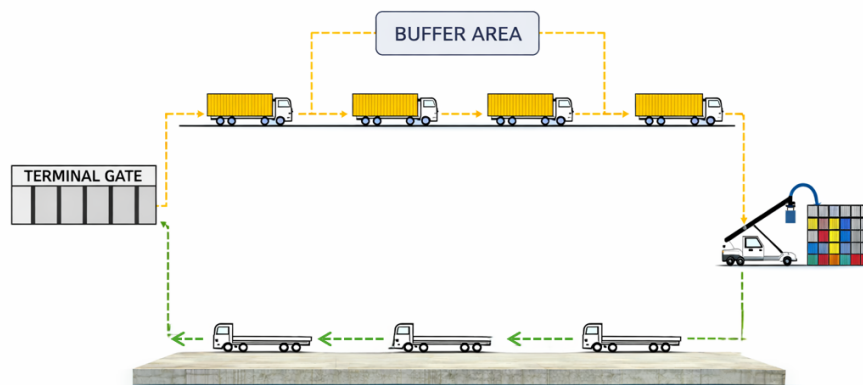


Figure 6.2: Inbound container flow via trucks

The truck reception cycle for export containers essentially consists of four main phases.

The first phase is the entry of the truck carrying the export container through the terminal gate. Before reaching the gate, the truck driver has confirmed the booking at the cargo office located outside the gate. Once the container is authorized for delivery to the terminal, the driver receives a notification indicating that access to the terminal is permitted. Upon arrival at the gate, terminal operators register the truck license plate, the ISO code of the container, and the seal code applied to the container doors using a handheld device.

The second phase may follow three possible scenarios:

The first and most frequent scenario is that, after passing the gate, the truck proceeds directly to the assigned export block, where the container is unloaded.

The second scenario involves the container being subjected to a customs inspection by the 'Guardia di Finanza' before reaching the export area. This may occur due to routine random checks, missing required documentation, or specific alerts issued by the relevant authorities.

The third scenario requires the truck to remain temporarily in a parking area within the terminal,



Figure 6.3: Loading process of containers onto the vessel

adjacent to the container yards, known as the Buffer Area. To regulate the inbound truck flow, the terminal employs a Truck Management System (TMS). This system monitors the number of trucks currently inside the terminal, verifies how many and which trucks are already present in the export blocks, and based on the capacity of the blocks to receive additional trucks-decides whether incoming trucks may proceed to their destination block or must wait in the buffer area until congestion is reduced.

The third phase consists of unloading the container from the truck onto the container stacks using a Reach Stacker (RS). This operation takes place in the export blocks (with the exception of reefer and IMO containers), which are therefore exclusively served by RS equipment. The selection of the export block and the exact stacking position of the container depends on several factors, including the container weight class, container type, the vessel onto which it will be loaded, and the port of destination. These criteria are defined to comply with vessel stowage plans.

The fourth phase is the return of the truck from the export block to the exit gate. In some cases, once the truck has unloaded the export container, it may move to the import areas to be reloaded with another container. In practice, this allows both truck reception and truck delivery operations to be performed within the same cycle and using the same vehicle. This approach is particularly advantageous for haulage companies, as it reduces empty movements (non-productive trips), and it also benefits the terminal by decreasing the number of trucks that need to be handled separately for container pickup operations in the future.

Following the arrival of containers at the terminal and their temporary storage in the yard, the process of loading containers onto the vessel is initiated.

Container transport operations from the yard to the vessel require the coordinated use of at least three types of equipment. Standard containers stacked in the export blocks are first retrieved by the RS and loaded onto a terminal Prime Mover (PM). The PM then transports the container from the export block to the quay crane working area, positioning it beneath the crane legs.

The Quay Crane (QC) is responsible for lifting the container from the PM and loading it on-board the vessel at a specific location defined by the vessel stowage plan, which assigns precise coordinates in terms of Bay, Row, and Tier.

Vessel loading operations typically start immediately after the completion of the discharge operations for the same vessel. For this reason, in operational reports these volumes are accounted for as outbound volumes of the vessel cycle.

In addition to containerized cargo, vessel loading operations may also include break-bulk cargo. This category refers to goods whose dimensions or characteristics prevent them from being handled as standard or out-of-gauge containers and therefore require special handling procedures, often involving external specialized companies. Such cargo may include particularly heavy items, oversized loads, or goods of high value.

6.1.2 Artificial neural network predictive model

As discussed in the objectives of this study, one of the main goals is the prediction of container flows entering and leaving the terminal by truck. To this end, the needs and requirements of the terminal management were first analyzed in order to clearly define the scope and objectives of the work. The problem under investigation was identified through discussions with industry experts and port operators.

Based on the results of this initial analysis, a neural network based methodology was selected for predicting container arrivals. The input and output datasets for the neural network were then defined accordingly. The neural network was trained and validated using the Leave-One-Out Cross-Validation (LOOCV) approach within the MATLAB environment. Finally, a post-processing technique was applied to further refine and improve the prediction results obtained from the neural network.

One of the most critical aspects of the predictive model is the definition of its input variables. These inputs are closely related to vessel operations and are described in detail below. The model output consists of the predicted number of containers. In the MATLAB implementation of the neural network, the data are organized into an input matrix containing the input features and an output matrix containing the corresponding target values. Furthermore, the relationship between the input matrix and the output matrix, which represents the variable to be predicted, is discussed.

The key features $\underline{x} = \{x_{ik}, \quad i = 1.., N \quad k = 1, ..70\}$ for each ship i , where N refers to the maximum number of ships in the dataset, are:

- *Service identification number* - x_{i1} : A service represents the repeated trip made by a group of ships, with a periodicity of one week, meaning that each week a ship of each service will visit the terminal at an agreed time window. These services vary in terms of the number of movements, signifying the number of containers they transport in the terminal, their arrival windows at the terminal, and whether or not they respect those schedules.

In this thesis, data pertaining to "on-spot services" (i.e., services whose ships do not have a time window to respect) are not included.

- *Delay* - x_{i2} :

$$x_{i2} = \begin{cases} 1 & \text{if } i\text{-th ship was delayed} \\ 0 & \text{otherwise} \end{cases}$$

Ships may or may not respect their contractual arrival window at the terminal, which can affect the arrival curve of containers by truck for the specific ship. This impact can be seen by the increased amount of time available for accepting containers meant for that specific ship.

- *Delay cause* - x_{i3} :

$$x_{i3} = \begin{cases} 1 & \text{if the cause of delay was the } i\text{-th ship itself} \\ 0 & \text{otherwise} \end{cases}$$

The start of the operational activities of a ship may be delayed due to two principal reasons: because of issues associated with the ship itself or due to the terminal inability to begin the ship operations on time. These two scenarios present distinct operational situations within the context of port processes.

- *Delay amount* - x_{i4} : The delay is calculated as the difference between the actual arrival time of the ship and its scheduled arrival and it is expressed in hours.
- *Yard Opening - Yard Closing Interval* - x_{i5} : It is the designated time period during which the terminal yard is open to receive containers delivered by trucks for export on a specific scheduled ship.
- *Year* - x_{i6} : The year of ship arrival.
- *Month* - x_{i7} : The month of ship arrival.
- *Containers* - x_{i8} : An integer variable indicating the planned number of containers arriving at the port by truck and expected to be exported with each ship.

As part of the input matrix for the ANN, two additional groups of variables are also included:

- *Holidays*: $x_{i9}-x_{i39}$ The objective is the one of predicting the number of containers that will arrive in the terminal every day within the interval from Yard Opening time to Yard Closing time ($[YO, YC]$). To achieve this, a mask to identify Sundays and holidays falling within this interval, that is, days in which trucks do not transport containers to the terminal, has been included.
- *Days of the week*: $x_{i40}-x_{i70}$ With the same logic as before, in this case, another mask with numbers from 1 to 7 to indicate the day of the week has been used. 1 for Monday and 7 for Sunday.

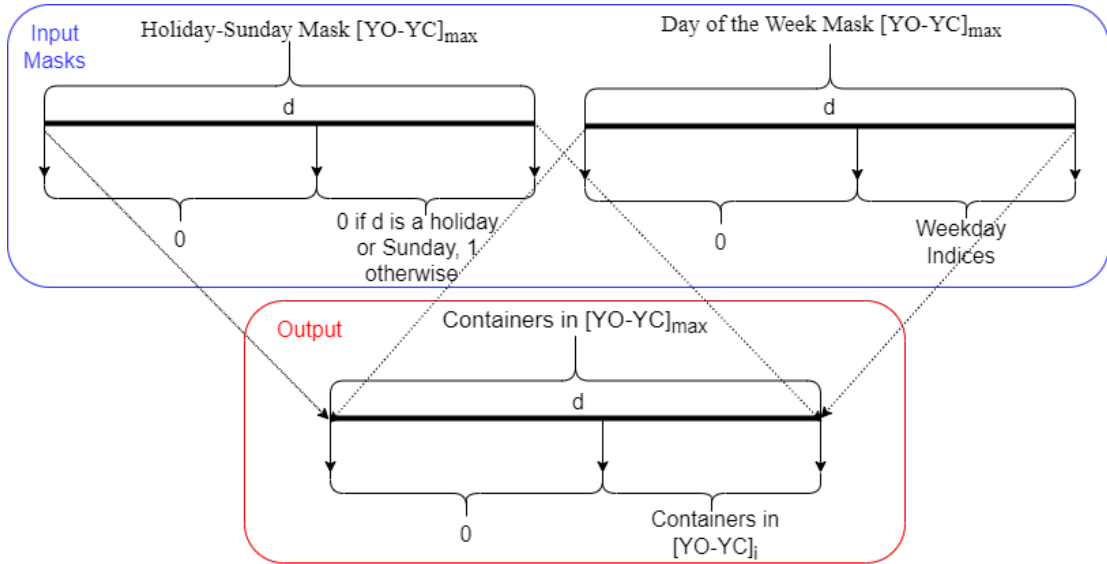


Figure 6.4: Input masks and output relationship

As shown in Fig. 6.4, the output matrix of the predicted model represents the container distribution within the time interval $[YO, YC]_i$ for a specific ship, denoted as i . To ensure uniformity, the largest interval among the ships, referred to as $[YO, YC]_{max}$, has been chosen as the standard one. If we denote T_{max} as the length of the maximum interval and T_i as the length of an interval i , then the container distribution within the $[YO, YC]_{max}$ interval for the $[YO, YC]_i$ interval begins from index $(T_{max} - T_i) + 1$ and extends to the end. The remaining indices are filled with zeros. This same principle is applied to the two masks present in the input matrix. It is important to note that the day following the end of the interval aligns with the ship's arrival day.

Based on the previously described input and output features, the Artificial Neural Network (ANN) adopted in this study is implemented as a feed-forward neural network architecture. The

network consists of an input layer with 70 neurons, corresponding to the defined input features, a single hidden layer with 10 neurons, and an output layer comprising 31 neurons. The hidden layer employs the tansig transfer function as its activation function, while the output layer uses the purelin transfer function to produce continuous-valued predictions. The mathematical formulations of these transfer functions are presented below.

$$\varphi(z) = \text{tansig}(z) = \frac{2}{1 + e^{-2z}} - 1 \quad (6.1)$$

and,

$$\varphi(z) = \text{purelin}(z) = z \quad (6.2)$$

6.1.2.1 Leave-one-out cross-validation

In this study, the LOOCV method, a well-established resampling technique used for model validation, has been used. Each iteration of LOOCV involves systematically leaving out one data point from the dataset while training the model on the remaining data. The model is tested in the data point left out, and the Mean Squared Error (MSE) and correlation index (R) are calculated. This process is repeated n times (where n is the total number of observations in the dataset), and then the MSE is calculated as the average of each MSE_j and the R to be the average of all of the R_j :

$$\text{MSE} = \frac{1}{n} \sum_{j=1}^n \text{MSE}_j \quad (6.3)$$

$$\text{R} = \frac{1}{n} \sum_{j=1}^n R_j \quad (6.4)$$

, where n is the number of data samples. The performance of the ANN can be evaluated through this process.

6.1.2.2 Post-processing procedure

After obtaining the results from the application of the ANN, a post-processing procedure is applied to increase the model prediction capabilities. This procedure includes:

1. Forcing negative predictions to zero
2. Zeroing predictions before the yard opening date and after the yard closing date
3. Zeroing predictions on holidays and Sundays. The quantities of containers predicted on holidays and Sundays are distributed equally on the days allowed within the interval $[YO, YC]$

4. Normalization process according to the total planned container number

6.1.3 Heuristic predictive model

The results of the predictive model are then compared with those obtained from a heuristic model, which is described below. The heuristic procedure is defined through the steps presented in Algorithm 3.

Algorithm 1 Container prediction Algorithm

Require: Take the i -th ship.

- 1: Set to zero the number of containers related to Sunday and holiday days.
- 2: Take the k -th time interval $[Tl, Tu]_{i,k}$, which is the time interval between two holidays inside the $[YO, YC]_i$, where $k = 1, ..M$ and M is the number of $[Tl, Tu]_{i,k}$ intervals inside $[YO, YC]_i$.
- 3: Compute the mean and variance of the interval as

$$\mu_{i,k} = \frac{(Tu_{i,k} - Tl_{i,k})}{2}$$

$$\sigma_{i,k}^2 = |Tu_{i,k} - Tl_{i,k}|$$

- 4: Approximate the containers in days, inside the $[Tl, Tu]_{i,k}$, with a Gaussian distribution function having the above mean and variance.
 - 5: Normalize the number of containers in $[YO, YC]_i$ relative to the total planned number of containers for ship i .
 - 6: Steps 2-4 are repeated for each k -th interval time.
-

Fig.6.5 shows the distribution of containers in the $[YO, YC]_{max}$ for the i -th ship. It illustrates the $[YO, YC]_i$ interval, starting from YO until YC as indicated in the graph, and the $[Tl, Tu]_{i,k}$ time intervals between two vacations inside the $[YO, YC]_i$.

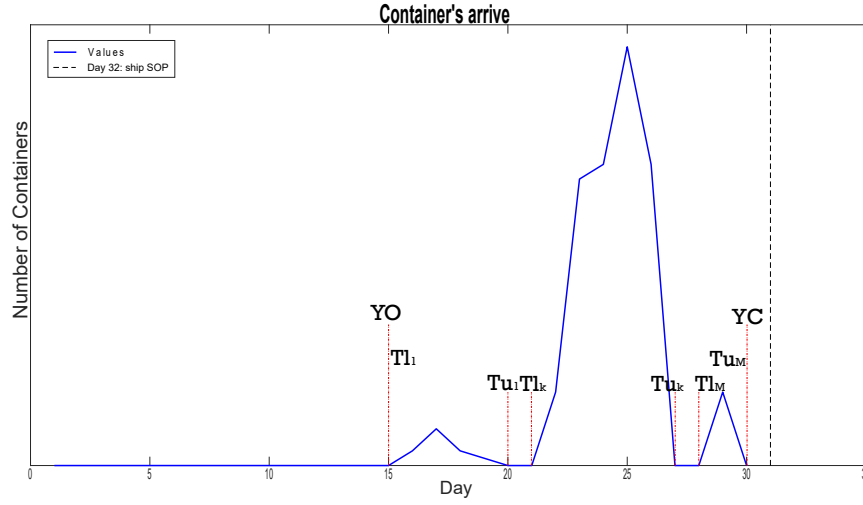


Figure 6.5: Illustrative graph of time intervals for a ship i

6.1.4 Export data description

The dataset considered in this study covers the period from January 2021 to July 2023. The data related to the number of containers were previously normalized in order to account for business restrictions. The dataset includes only containers that entered the terminal by truck and subsequently left by vessel, thereby completing an export cycle.

Specific container categories were excluded from the dataset, as they follow different operational procedures. These include special containers such as those carrying dangerous goods (IMO), out-of-gauge (OOG) containers, and refrigerated containers. Containers that experienced a change of voyage (COV) were also excluded. COV containers initially arrive at the terminal for export on a designated vessel but, due to circumstances such as vessel delays or customs documentation issues, are eventually exported on another vessel within the same service.

Table 6.1 presents selected characteristics of the input features, including their type, mean value, standard deviation (SD), and range.

During the data analysis phase, cases deviating from typical operational behavior were identified. These anomalies introduced additional complexity in detecting consistent patterns and correlations within the dataset.

One of the key variables influencing truck arrivals is the number of days during which the yard remains open to receive export containers for a specific vessel. Under standard operating conditions, this period is expected to be seven days. For vessels that are not delayed, the yard opening period would generally not exceed this duration. However, the analysis revealed instances that deviated from this assumption.

For example, even in cases where a vessel was not classified as delayed, the yard opening period was observed to be either longer than seven days or significantly shorter. Although these situations can be justified from an operational perspective, they occur with sufficient frequency that they cannot simply be treated as outliers and removed from the dataset. Instead, the objective is to demonstrate that the proposed methodology is capable of reliably predicting such scenarios as well.

From an operational standpoint, several factors may explain these deviations. If the first vessel in a service experiences a delay, its yard opening period is extended accordingly. The subsequent vessel in the same service, even if arriving on schedule, may berth immediately after the delayed vessel and therefore have a substantially reduced yard opening period for export containers. According to contractual agreements, the terminal accepts export containers for only one vessel per service at a time. Consequently, situations arise in which the second vessel's yard opening period is considerably shorter than seven days.

In other cases, a vessel may arrive within its contractual time window, measured in hours, but during the following week. This may occur when the subsequent vessel in the service is also delayed. Although the vessel is contractually considered "on time," the yard opening period may still exceed seven days. In such circumstances, the vessel is classified as not delayed, despite the extended yard availability period.

Table 6.1: Input features for the export container prediction model

Feature	Type	Mean	SD	Range
x_{i1}	Categorical	-	-	[1, 13]
x_{i2}	Binary	-	-	{0, 1}
x_{i3}	Binary	-	-	{0, 1}
$x_{i4}(\text{hours})$	Numerical	56.1225	73.3671	[-619.68, 768.24]
$x_{i5}(\text{days})$	Numerical	8.6681	4.1222	[2, 31]
x_{i6}	Categorical	-	-	[2021, 2023]
x_{i7}	Categorical	-	-	[1, 12]
x_{i8}	Numerical	-	-	-
x_{i9} to x_{i39}	Binary	-	-	{0, 1}
x_{i40} to x_{i70}	Categorical	-	-	[0, 7]

The overall dataset is composed of 928 ships from 13 distinct services that arrived at the PSA GP terminal during the specified period. Table 6.2 presents some statistics for each service, including the total number of ships, the count of delayed arrivals, the years in which ships from each service arrived, and the average length of the ships. 39.44% of the ships arrived in 2021, 35.67% of them during 2022 and 24.89% during the current year. 72.6% of the them where delayed, meanwhile the other part respected their time window.

Table 6.2: Statistics for each service (export containers)

Service	Ships	Delayed	Years Of Ops	Avg. Length
$x_{i1} = 1$	24	20	2021	263.5
$x_{i1} = 2$	84	65	2021, 2022, 2023	367.4
$x_{i1} = 3$	23	13	2021	367.2
$x_{i1} = 4$	3	3	2021	367.6
$x_{i1} = 5$	38	28	2022, 2023	196.1
$x_{i1} = 6$	97	78	2021, 2022, 2023	363.6
$x_{i1} = 7$	97	63	2021, 2022, 2023	365.5
$x_{i1} = 8$	107	84	2021, 2022, 2023	301.2
$x_{i1} = 9$	90	54	2021, 2022	273.5
$x_{i1} = 10$	103	78	2021, 2022, 2023	332.7
$x_{i1} = 11$	121	80	2021, 2022, 2023	155.8
$x_{i1} = 12$	27	16	2023	304.1
$x_{i1} = 13$	114	92	2021, 2022, 2023	318.9

6.1.5 Export prediction results

After implementing both the predictive model and the heuristic algorithm, the results of the two methodologies are obtained. From the prediction model based on the application of the neural network, the results show an average R index of $0.8304 \in [-0.1027, 0.9988]$ and an average MSE of $0.0022 \in [1.7848 \cdot 10^{-5}, 0.0448]$. In the case of the heuristic algorithm, the results show an average R index of $0.6083 \in [0.0715, 0.9948]$ and an average MSE of $0.0045 \in [1.0819 \cdot 10^{-4}, 0.0312]$.

In 83.08% of the cases the model based on ANN predicts the arrival curve of the containers with an index R greater than 0.7. Meanwhile for 67.13% of the cases the value of R is greater than the average. For 68.32% of the cases the MSE value is less or equal than the average one. In the case of the heuristic algorithm only in the 54.3% of the cases the index R is greater than 0.7. Meanwhile for 51.72% of the cases the value of R is greater than the average. For 60.02% of the cases the MSE value is less or equal than the average one.

Given that the LOOCV methodology is employed as part of the predictive model, it allows to inspect the ANN results for each sample used for testing in the dataset, enabling the identification of cases where the results are not satisfactory. In 16.92% of the cases, where $R \leq 0.70$, 75.8% of them feature an interval $[YO, YC]$ greater than 7. And, in 32.87% of the cases where R is less than the average, 68.52% of them feature an interval $[YO, YC]$ greater than 7. A large interval can lead to a distribution that changes significantly from the others, making it challenging to predict this curve with high accuracy. In the subset with an interval length $[YO, YC]$ less than 7, corresponding to 50 cases, 68% of them have a total planned containers very inferior to the others, while the the average number of planned containers across all cases is approximately four

times larger. Only 16% of the total cases have such a small number of containers, and 55% of them are part of the group with not good results in terms of R . This small number of containers can limit the neural network's ability to produce good results. It is important to note also that, the average MSE and R for each dataset used for training, are very similar to the average MSE and R of testing, indicating that the prediction model isn't overfitting and is generalizing well. The graphs below illustrate the results of both the ANN and the heuristic approach for three cases in the dataset.

Fig. 6.6 shows one of the examples where the predictive model based on ANN effectively predicted the curve of containers with an MSE of 0.0003 and an R index of 0.9766. Meanwhile the heuristic approach results for the same case, indicate an MSE of 0.0018 and an R of 0.8386, values that show a performance inferior to that of the ANN. Regarding the features of this case, we are dealing with a ship that was delayed and had an interval $[YO, YC]$ of 6 days. In Fig. 6.7 it is shown a case when the ANN performed less effectively with an MSE of 0.001 and R of 0.84. The the analog graph for the heuristic case shows an MSE of 0.0024 and R of 0.5622. In this case, the ship experienced a delay, resulting in a $[YO, YC]$ interval of 14 days. This long number of days, results in a more complex curve of container arrival making it more difficult for the ANN to predict it with high accuracy.

Table 6.3 summarizes the results obtained using the ANN and heuristic approaches.

Table 6.3: Comparison of ANN and Heuristic results

	ANN			Heuristic
	Training	Testing	Testing + Post Procedure	
Average MSE	0.0192	0.0192	0.0022	0.0045
Average R	0.7926	0.7925	0.8304	0.6083

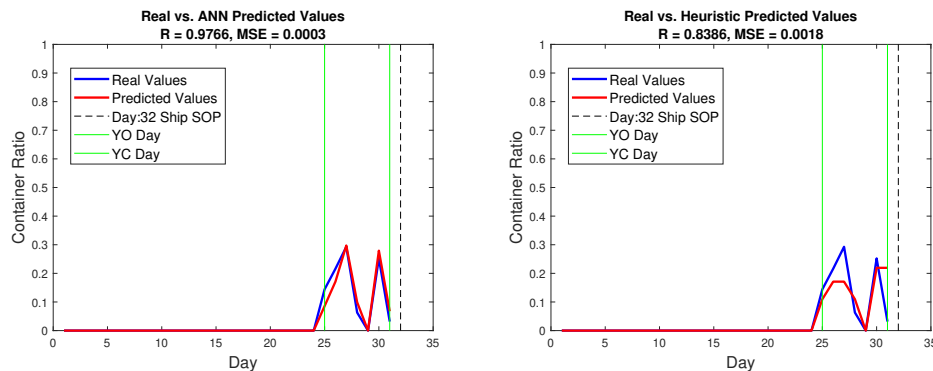


Figure 6.6: Actual vs. ANN and Heuristic predicted container arrival curve (Ship 149)

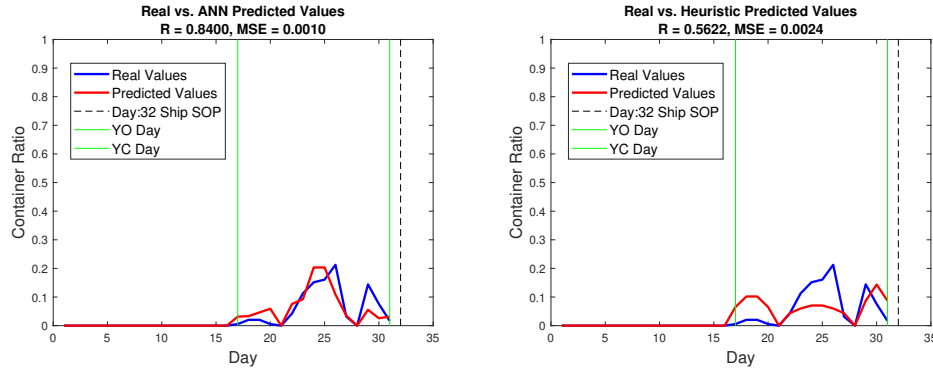


Figure 6.7: Actual vs. ANN and Heuristic predicted container arrival curve (Ship 183)

6.2 Prediction of import containers via truck

Similarly to the previous section, this section focuses on the prediction of import containers transported by truck and presents the data and results associated with the proposed methodology. It starts with a brief overview of the process, followed by a detailed description of the predictive model and the dataset employed. The section concludes with a discussion of the obtained results.

6.2.1 Brief reminder of the import process

As in the case of container export, a brief overview of the container import process is presented. This process starts with the discharge of containers from the vessel (Fig. 6.8) and ends with their collection by trucks at the terminal (Fig. 6.9).

Vessel discharge operations begin with the unloading of containers from the vessel by the QC. Each container is lifted from the ship and placed onto a PM, which transports it from the quay to the yard.

Once in the yard, the container is discharged from the PM and stacked within the designated import block. For full import containers, this operation is carried out by RTGs, which retrieve the container from the PM and position it within the stack. Empty containers, on the other hand, are not handled by RTGs but are instead managed in dedicated areas served by RS.

After an import block has been fully populated, it is opened to external truck arrivals. Trucks entering the terminal access the import areas to collect containers. In this phase, the RTG retrieves the container from the import block and loads it directly onto the truck, thereby completing the container import cycle.



Figure 6.8: Discharging process of containers from the vessel

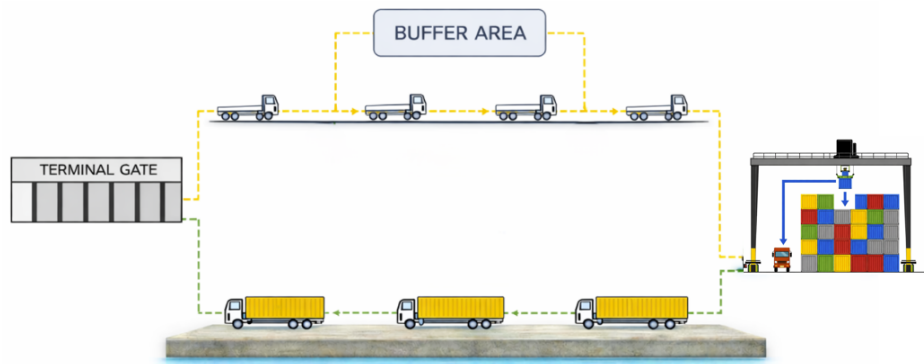


Figure 6.9: Outbound container flow via trucks

6.2.2 Artificial neural network predictive model

For the prediction of import containers, a neural network architecture similar to that adopted for the export case is employed, with some differences in the definition of the main input features, as described below. In this case, the target variable corresponds to the number of TEUs rather than the number of container units, as considered in the export prediction. This choice does not alter the structure of the dataset; it is motivated solely by the availability of import data expressed in

TEUs.

The input features are defined for each ship, with the objective of predicting the number of TEUs associated with a specific ship and subsequently leaving the terminal by truck.

The dataset used for training the ANN is structured in an input matrix denoted by $\underline{x}$, whose elements are x_{ik} , with $k = 1, \dots, K$ and $i = 1, \dots, N$, where K represents the total number of features (equal to 23 in this study) and N denotes the number of observations.

All input features are related to quay operations and are described as follows:

- *Service code* – x_{i1} : A service refers to the regular rotation of a group of ships, typically operating on a weekly basis. Each week, one ship from a given service is scheduled to arrive at the terminal within a predefined time window. Services differ in terms of transported TEU volumes, scheduled arrival times, and adherence to those schedules.

- *Delay* – x_{i2} :

$$x_{i2} = \begin{cases} 1 & \text{if the } i\text{-th ship is delayed} \\ 0 & \text{otherwise} \end{cases}$$

Ships do not always arrive within their contractual time window. When this occurs, operations are postponed, which may influence the timing of truck arrivals for import TEU pick-up.

- *Delay cause* – x_{i3} :

$$x_{i3} = \begin{cases} 1 & \text{if the delay is caused by the ship} \\ 0 & \text{otherwise} \end{cases}$$

Delays may originate either from the ship itself or from terminal operational constraints that prevent the ship from being handled on time. These two situations have different operational implications.

- *Delay amount* – x_{i4} : The magnitude of the delay, expressed in hours, defined as the difference between the actual arrival time and the scheduled arrival time of the ship.
- *Year* – x_{i5} : The calendar year of the ship's arrival.
- *Month* – x_{i6} : The calendar month of the ship's arrival.
- *Planned TEUs* – x_{i7} : The planned number of TEUs expected to arrive with the specific ship and to be subsequently collected from the terminal by trucks.

- *Holiday indicators* – $x_{i8}-x_{i15}$: For each day within the Free Interval (FI), starting from the Import Container Ready Date (ICRD), a binary variable is included in the input matrix. The variable is set equal to 0 if the day is a holiday or Sunday, and 1 otherwise. The FI typically spans seven days from the ICRD, during which import containers may remain in the terminal without incurring storage charges. After this period, storage fees apply.
- *Day-of-week indicators* – $x_{i16}-x_{i23}$: For each day within the FI, an additional variable is included indicating the day of the week, taking integer values from 1 to 7, where 1 corresponds to Monday.

As for the output matrix, for each vessel, it contains the number of TEUs imported for each day of FI. Meanwhile, the last day, respectively the ninth day, contains the sum of all the TEUs being imported after the FI is finished. The output matrix for the ANN is denoted by $\underline{o}$, whose components are o_{ip} , with $p = 1, \dots, P$, and $i = 1, \dots, N$, where $P = 9$ is the number of features across N observations.

Similarly to the export case, the LOOCV technique is employed for training and validating the prediction model. In addition, a post-processing procedure, analogous to the one applied in the export case, is proposed to further refine the ANN results. The algorithm describing this procedure is presented in the following subsection.

6.2.2.1 Post-processing procedure

In order to improve the outcomes of the neural network, a post-processing (PP) algorithm is proposed, as shown in Algorithm 2, in which:

- v indicates the vessel (ship).
- s represents the sum of TEUs inaccurately predicted during holidays or Sundays by ANN.
- FI_v denotes the free interval associated with vessel v .
- nr indicates the number of days within FI_v , that are either holidays or Sundays.
- d refers to the days within FI_v .
- TEU_d indicates the number of TEUs predicted by the model in each day d .
- TEU_9 refers to the total TEUs predicted to arrive after FI_v .

Algorithm 2 Post-Processing Algorithm

```
1: for each vessel  $v$  do
2:    $s \leftarrow 0$ 
3:    $nr \leftarrow 0$ 
4:   for  $d \in FI_v$  do
5:     if  $TEU_d < 0$  then
6:        $TEU_d \leftarrow 0$ 
7:     end if
8:     if  $d$  is a holiday or  $d$  is a Sunday then
9:        $s \leftarrow s + TEU_d$ 
10:       $nr \leftarrow nr + 1$ 
11:       $TEU_d \leftarrow 0$ 
12:    else
13:       $TEU_d \leftarrow TEU_d$ 
14:    end if
15:  end for
16:  for  $d \in FI_v$  do
17:    if  $d$  is not a holiday or  $d$  is not a Sunday then
18:       $TEU_d \leftarrow TEU_d + \frac{s}{|FI_v| - nr}$ 
19:    end if
20:  end for
21:  if  $TEU_9 < 0$  then
22:     $TEU_9 \leftarrow 0$ 
23:  end if
24:  Normalize the values in each day according to the total planned number
25: end for
```

6.2.3 Import data description

The dataset includes information on trucks arriving to collect containers that were previously discharged from ships, covering the period from January 2021 to July 2023. In total, the dataset comprises records for 901 ships operating across 13 different services.

Table 6.4 presents selected characteristics of the input features, including their type, mean value, standard deviation, and range. In addition, Table 6.5 reports service-level statistics, such as the total number of ships per service, the number of delayed ships, the years during which each service was active within the study period, and the average ship length.

A preliminary analysis was conducted to examine the main characteristics of the dataset. As observed, Services 1, 5, and 11 are associated with smaller ships, which consequently results

in lower TEU transport capacity. With regard to delays, approximately 74% of the ships in the dataset experienced a delay. It should be noted that the study period overlaps with the COVID-19 pandemic, which had a significant impact on maritime logistics and contributed to the elevated proportion of delayed ships.

On average, 20% of the TEUs were collected outside the Free Interval (FI).

The data related to the planned number of TEUs for each vessel is not presented due to privacy restrictions from the terminal.

Table 6.4: Input features for the import container prediction model

Feature	Type	Mean	SD	Range
x_{i1}	Categorical	-	-	[1, 13]
x_{i2}	Binary	-	-	{0, 1}
x_{i3}	Binary	-	-	{0, 1}
$x_{i4}(\text{hours})$	Numerical	59.39	70.58	[-149.76, 768.24]
x_{i5}	Categorical	-	-	[2021, 2023]
x_{i6}	Categorical	-	-	[1, 12]
x_{i7}	Numerical	-	-	-
x_{i8} to f_{i15}	Binary	-	-	{0, 1}
x_{i16} to f_{i23}	Categorical	-	-	[1, 7]

Table 6.5: Statistics for each service (import containers)

Service	Vessels	Delayed	Years Of Ops	Avg. Length
$x_{i1} = 1$	26	22	2021	263.5
$x_{i1} = 2$	87	68	2021, 2022, 2023	367.4
$x_{i1} = 3$	10	6	2021	365.6
$x_{i1} = 4$	15	14	2021	367.8
$x_{i1} = 5$	18	15	2022, 2023	204.6
$x_{i1} = 6$	97	78	2021, 2022, 2023	317.9
$x_{i1} = 7$	97	63	2021, 2022, 2023	363.5
$x_{i1} = 8$	108	86	2021, 2022, 2023	365.8
$x_{i1} = 9$	87	51	2021, 2022	301.4
$x_{i1} = 10$	104	81	2021, 2022, 2023	333.2
$x_{i1} = 11$	118	78	2021, 2022, 2023	156.6
$x_{i1} = 12$	24	16	2023	373.8
$x_{i1} = 13$	110	87	2021, 2022, 2023	314.6

6.2.4 Import prediction results

Initially, the neural network was trained using the complete dataset of 901 ships, applying the LOOCV technique. The testing results corresponding to observations belonging to the same service were grouped, and the average MSE and average correlation coefficient R were calculated for each service.

Subsequently, the dataset was partitioned according to service, resulting in 13 separate subsets. The LOOCV procedure was then applied independently to each subset, and the corresponding average MSE and R values were computed. For each service, the results obtained from training on service-specific data were compared with those derived from training on the entire dataset.

For three services, improved performance was observed when the model was trained exclusively on observations associated with those services. In contrast, for the remaining services, the results in terms of MSE and R were slightly inferior compared to the model trained on the full dataset. An examination of the characteristics of these three services revealed that they differed in the planned number of import TEUs discharged, indicating that they correspond to smaller services, as discussed in the previous section.

Given their similarity in scale and operational characteristics, these three services were subsequently grouped together and used to train a unified model, thereby forming a larger and more representative dataset. Ultimately, two separate datasets were defined, categorized according to the planned number of TEUs to be transported.

Table 6.6: Predictive model results

	Dataset 1		Dataset 1 (OF)		Dataset 2	
	R	MSE	R	MSE	R	MSE
Train	0.7709	2.3244	0.7680	0.0257	0.7298	0.1375
Test	0.7714	3.3765	0.7618	0.0281	0.7330	0.2141
Test + PP	0.7850	0.0060	0.7779	0.0051	0.7495	0.0103

Dataset 1 is the largest, including data from 10 services, while Dataset 2 is created using data from the three excluded services. The results of training, testing, and the outcomes after applying the Post-Processing procedure are presented in Table 6.6. It is essential to note that the MSE is calculated for the normalized values of TEUs, with values for each day divided by the total planned TEUs for the vessel.

For Dataset 1, a considerable MSE is observed in both training and testing. Further investigation revealed that among the 739 cases used for testing, only 22 cases contributed to this high MSE. Upon reviewing these cases, it was identified that the total number of planned TEUs for these observations was less than 2% of the average planned number of TEUs for the entire Dataset 1. These observations can be considered anomalies in the dataset, as they do not align even

with the typical terminal operation perspective. Consequently, it was decided to exclude these 22 observations from the model, creating the Dataset 1 Outlier-Free (OF). The results of testing before and after the post-processing, with the exclusion of the 22 observations, are presented in Table 6.6, and they exhibit a clear improvement in terms of MSE.

For Dataset 1 (OF) the results of the predictive methodology show an average R index of 0.7779, where $R \in [-0.3196, 0.9938]$ and an average MSE of 0.0051, where $MSE \in [1.3343 \cdot 10^{-4}, 0.0742]$. Approximately 68.36% of the outcomes exhibit a higher value of R compared to the average, while approximately 71.69% of the results demonstrate a lower MSE value compared to the average. Meanwhile, in 62.76% of the cases, both a higher R value and a lower MSE value are observed compared to their respective averages. 37.52% of the cases have a value of R higher than 0.9.

Figs. 6.10a and 6.10b present two examples: one showcasing a very good performance of the predictive model, and the other where the values of R and MSE are close to the average.

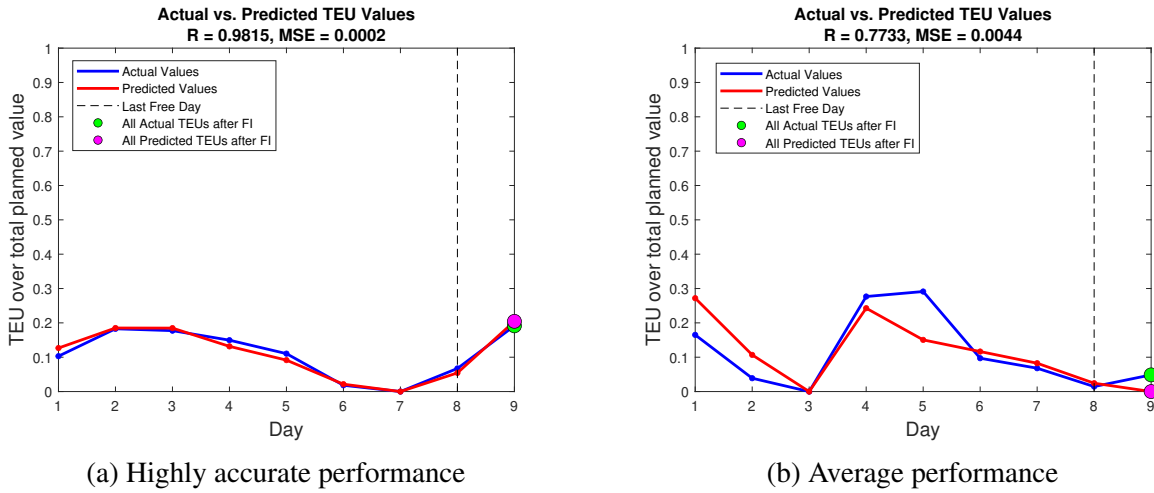


Figure 6.10: Actual vs. ANN predicted TEU arrival curves for Dataset 1 (OF)

In the case of Dataset 2, we applied the same approach. However, we could not identify a clear reason for the cases where the results were not as good. It is important to note that the smaller training set in this context makes it challenging for the ANN to capture the complexity of the underlying data relationships effectively. The results show an average R index of 0.7495, where $R \in [-0.2753, 0.9957]$ and an average MSE of 0.0103, where $MSE \in [4.0564 \cdot 10^{-4}, 0.1111]$. In 62.96% of the cases, the R value surpasses the average, while in 65.43% of the cases, the MSE is lower than the average. Only 51.8% of the cases have better values for both R and MSE compared to the averages. In 42.59% of the cases the value of R is higher than 0.9. In Figs. 6.11a and 6.11b, two examples from the second dataset are shown. In one, the ANN had a good result, and in the other, an average one.

As for the efficiency of the algorithm, it varies significantly depending on the dataset. For the

largest dataset (Dataset 1), the algorithm takes approximately 4.5 minutes to execute. In contrast, for the smaller dataset (Dataset 2), the execution time is less than one minute.

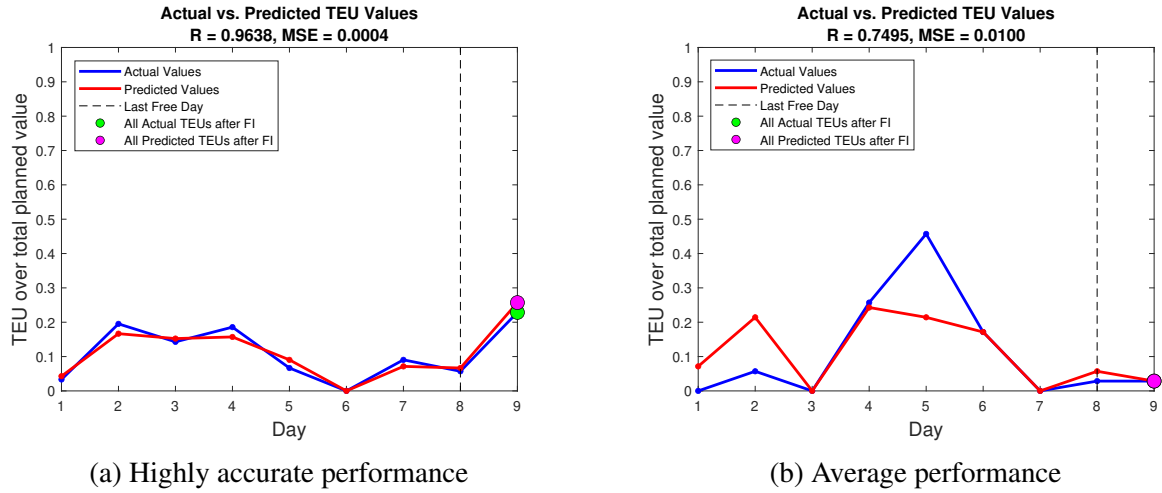


Figure 6.11: Actual vs. ANN predicted TEU arrival curves for Dataset 2

Table 6.7: A comparison with the Random Forest method

	ANN		Random Forest	
	R	MSE	R	MSE
Test	0.7618	0.0281	0.6309	0.0075
Test + PP	0.7779	0.0051	0.4907	0.0088

To compare the results and assess the efficiency of the proposed methodology, the Random Forest method was also applied using the TreeBagger function in MATLAB. The results presented in Table 6.7 correspond to the test and test plus PP phases for Dataset 1 (OF). As shown, the only advantage of the Random Forest method is its lower MSE during the test phase. However, when PP is applied, the MSE of the Random Forest method increases, resulting in poorer performance compared to the ANN.

Related publications. Parts of the work presented in this chapter have been published in [116].

Chapter 7

Truck Operations Optimization

This chapter focuses on the use of the predictions obtained in the previous chapter as inputs for the development of optimization models aimed at reducing congestion in specific areas of the terminal, as well as minimizing emissions. The chapter is divided into two main sections: congestion reduction and emission reduction, in which the corresponding mathematical models and results are presented.

7.1 Optimization of truck arrival patterns to reduce congestion

The prediction of container arrivals and container pick-ups via truck, described in the previous chapter, provides valuable decision-support information for terminal planners. In particular, it enables yard planners to anticipate potential peaks in yard occupancy and identify periods of critical congestion. Beyond its descriptive and forecasting value, this predictive framework also serves as the foundation for the optimization model developed in this chapter, which aims to mitigate congestion caused by concentrated truck arrivals within the terminal.

Based on the predicted number of import containers to be picked up on each day, the corresponding number of trucks required to perform these movements is estimated. This calculation accounts for operational characteristics such as the average proportion of trucks transporting a single container versus those transporting two containers. In this way, container flow predictions are translated into expected truck arrival flows, which constitute the main input to the optimization scheme.

The motivation behind the proposed optimization framework arises from the operational dynamics of high-volume container terminals. Large vessel calls generate substantial container

discharge volumes, which in turn create intense demand for truck pick-ups within specific Free Interval (FI) periods, as defined in the previous chapter. This concentration of truck arrivals during limited time windows often leads to high utilization of terminal areas and may induce congestion both at the gate and within yard operating zones.

To address this issue, a planning scheme for managing truck arrivals is proposed. The concept is inspired by Truck Appointment Systems (TAS), in the sense that it assumes the terminal operator has some degree of influence over the timing of truck arrivals. However, unlike conventional TAS implementations that aim to strictly allocate all truck visits to predefined time slots, the proposed approach does not seek to fully control or prescribe every truck movement. Instead, it focuses on selectively redistributing a portion of truck arrivals in order to smooth peaks, reduce congestion, and improve the overall balance of terminal workload.

For the implementation of the model, a specific month of predicted container pick-up data is selected as input. The optimization framework is structured in two sequential stages. The first stage addresses the redistribution of truck arrivals across days, aiming to balance daily workload levels. The second stage refines this allocation by distributing truck arrivals within each day across hourly time intervals. The detailed formulation of these two schemes is presented in the following subsections.

7.1.1 First phase of the optimization scheme: redistribution in days

The first phase of the optimization framework is aimed at preventing excessive daily peaks of truck arrivals within the selected planning horizon. The physical capacity and the available handling resources in each terminal area allow the definition of a threshold value, representing the maximum number of trucks that can be accommodated in that area within a single day. Since these capacity thresholds are defined separately for each terminal area, and each truck movement is associated with a specific destination area, the optimization model is formulated and solved independently for each area a , where $a = 1, \dots, A$, and A denotes the total number of terminal areas considered in the analysis.

V vessels discharging containers assigned to the area under examination are considered. This information is generally available to terminal management in advance, as container positioning across terminal areas is planned prior to vessel arrival. Given a planning horizon of D days, the parameters included in the formulation of the first-stage optimization problem, solved for each area a , are defined as follows:

- $b_v^{\text{FI}}, v = 1, \dots, V$ refers to the beginning of the free interval corresponding to vessel v , which indicates the first day on which trucks are allowed to reach the terminal to pick up import containers arrived with vessel v . Trucks aiming to pick up containers discharged from vessel v cannot reach the terminal before the b_v^{FI} date.

- $e_v^{\text{FI}}, v = 1, \dots, V$ refers to the end of the free interval corresponding to vessel v , indicating the final day of the interval during which trucks can pick up containers discharged from vessel v without incurring fees.
- $p_{v,d}, v = 1, \dots, V, d = 1, \dots, D$ is the predicted number of trucks expected at day d for vessel v picking-up containers from the considered area.

The predicted number of trucks assigned to a specific area on a given day is derived from two sources. First, the prediction methodology provides an estimate of the total number of trucks expected on that day. Second, operational information available from the terminal specifies the storage areas and the corresponding percentages of containers, associated with vessels belonging to a given service, that are allocated to each area.

- $\mathcal{D}_{\text{off}}$ is the set of days when the terminal is closed for trucks and, then, trucks cannot visit it. This includes Sundays and holidays.
- T_{max} is the maximum number of trucks that can arrive at the considered area in day d without causing congestion. T_{max} is computed as

$$T_{\text{max}} = Tr_{\text{max}} \times H$$

where Tr_{max} is the maximum number of trucks allowed in one hour, and H is the number of hours in which trucks can arrive in one day. As already mentioned Tr_{max} depends on the available space and handling resources present in the area.

Then, the problem is formulated as the redistribution of truck arrivals across the days of the planning horizon in order to minimize the occurrences in which the predefined threshold value T_{max} is exceeded. The decision variables included in the optimization model are defined as follows.

- $t_{v,d}^+, v = 1, \dots, V, d = 1, \dots, D, t_{v,d}^+ \in \mathbb{R}^+$, is the number of trucks added to the normal planning in day d for vessel v .
- $t_{v,d}^-, v = 1, \dots, V, d = 1, \dots, D, t_{v,d}^- \in \mathbb{R}^+$ is the number of trucks deducted to the normal planning in day d for vessel v .
- $t_{v,d}, v = 1, \dots, V, d = 1, \dots, D, t_{v,d} \in \mathbb{R}^+$ is the resulting number of trucks for vessel v in day d .

The statement of the optimization problem follows:

Problem 1

$$\min \sum_{d=1}^D \max \left(\sum_{v=1}^V t_{v,d} - T_{max}, 0 \right) \quad (7.1)$$

subject to

$$t_{v,d} = p_{v,d} + t_{v,d}^+ - t_{v,d}^- \quad v = 1, \dots, V \quad d = 1, \dots, D \quad (7.2)$$

$$\sum_{d=1}^D t_{v,d} = \sum_{d=1}^D p_{v,d} \quad v = 1, \dots, V \quad (7.3)$$

$$\sum_{v=1}^V t_{v,d}^- \leq \max \left(\sum_{v=1}^V p_{v,d} - T_{max}, 0 \right) \quad d = 1, \dots, D \quad (7.4)$$

$$t_{v,d}^+ = t_{v,d}^- = 0 \quad v = 1, \dots, V, d \in \mathcal{D}_{off} \quad (7.5)$$

$$t_{v,d}^+ = t_{v,d}^- = 0 \quad v = 1, \dots, V, d < b_v^{FI} \quad (7.6)$$

$$t_{v,d}^+ = t_{v,d}^- = 0 \quad v = 1, \dots, V, d > e_v^{FI} \quad (7.7)$$

The objective function minimizes the total number of trucks exceeding the threshold value T_{max} .

Constraints (7.2) determine the number of trucks assigned to the area for the pick-up of containers associated with vessel v on day d . Constraints (7.3) guarantee the conservation of the total number of trucks for each vessel after the redistribution process. Constraints (7.4) impose that the maximum reduction in the number of trucks assigned to the area on day d cannot exceed the number of trucks that surpass the threshold in the absence of redistribution. This condition ensures that only the minimum necessary modifications are applied to the predicted truck arrivals. Finally, constraints (7.5), (7.6), and (7.7) ensure that truck arrivals are not reassigned to days when the terminal is closed or to days outside the admissible time interval $[b_v^{FI}, e_v^{FI}]$ defined for each vessel v .

Problem 1 is further simplified by linearizing the cost function, as in the following formulation.

Problem 2

$$\min \sum_{d=1}^D C_d \quad (7.8)$$

subject to (7.2), (7.3), (7.4), (7.5), (7.6), (7.7), and

$$C_d \geq \sum_{v=1}^V t_{v,d} - T_{max} \quad d = 1, \dots, D \quad (7.9)$$

$$C_d \geq 0 \quad d = 1, \dots, D \quad (7.10)$$

7.1.2 Second phase of the optimization scheme: redistribution along hours of the day

The second phase of the optimization framework is also formulated separately for each terminal area, since the areas can be treated as operationally independent. Starting from the optimized daily distribution of trucks, which guarantees the minimum feasible deviation with respect to the daily threshold, a further optimization problem is defined for each area a and each day d . This second phase is required because truck arrivals within a day are not uniformly distributed across hours and are not directly controlled by terminal management. Consequently, even if daily thresholds are satisfied, hourly peaks may still occur, potentially exceeding the hourly capacity of the area and generating congestion.

Therefore, the objective of this phase is to optimize the intra-day distribution of predicted truck arrivals across hourly time intervals in order to prevent violations of the hourly threshold Tr_{max} . The optimization problem is thus formulated and solved independently for each combination of day d and area a .

In this phase, the total number of trucks assigned to area a on day d , originating from all vessels v , is aggregated. These daily totals are then distributed across the hours of the day according to the recurrent historical pattern of truck arrivals observed for the considered area.

The parameters included in the formulation of this optimization problem are defined as follows:

- $p_h, h = 1, \dots, H$ is the predicted number of trucks expected at hour h in the considered area. The total number in one day is the output of the first phase of the optimization scheme. For the distribution across hours, an average curve of truck arrivals is selected for each service. This curve represents the distribution of truck arrivals throughout the H-hour period.
- Tr_{max} represents the threshold for trucks, indicating the maximum allowable number of trucks that can arrive in the area during one hour without resulting in congestion.

The decision variables included in the optimization problem are the following.

- $t_h^+, h = 1, \dots, H \in \mathbb{R}^+$, is the number of trucks added to the normal planning in hour h .
- $t_h^-, h = 1, \dots, H \in \mathbb{R}^+$ is the number of trucks deducted to the normal planning in hour h .
- $t_h, h = 1, \dots, H \in \mathbb{R}^+$ is the resulting number of trucks for hour h .

The statement of the optimization problem follows:

Problem 3

$$\min \sum_{h=1}^H \max(t_h - Tr_{max}, 0) \quad (7.11)$$

subject to

$$t_h = p_h + t_h^+ - t_h^- \quad h = 1, \dots, H \quad (7.12)$$

$$\sum_{h=1}^H t_h = \sum_{h=1}^H p_h \quad (7.13)$$

$$t_h^- \leq \max(p_h - Tr_{max}, 0) \quad h = 1, \dots, H \quad (7.14)$$

The objective function in Problem 3 minimizes the total number of trucks exceeding the hourly threshold Tr_{max} .

Constraints (7.12) determine the number of trucks arriving at the terminal for container pick-up during hour h . Constraints (7.13) guarantee the conservation of the total daily number of trucks after the redistribution process.

Constraints (7.14) impose that the maximum reduction in the number of trucks assigned to hour h cannot exceed the number of trucks that surpass the threshold in the absence of redistribution. This condition ensures that the redistribution introduces only the minimum necessary adjustments to the predicted hourly arrivals.

Problem 3 is subsequently simplified through linearization of the objective function, leading to the following equivalent formulation.

Problem 4

$$\min \sum_{h=1}^H K_h \quad (7.15)$$

subject to (7.12), (7.13), (7.14), *and*

$$K_h \geq t_h - Tr_{max} \quad h = 1, \dots, H \quad (7.16)$$

$$K_h \geq 0 \quad h = 1, \dots, H \quad (7.17)$$

7.1.3 Results of the optimization schemes

The analysis of the proposed optimization framework focuses on the month of May 2023. For this period, daily predictions are considered for each individual vessel, as described in the previous chapter. Since the forecasting methodology provides estimates on a per-vessel and per-day basis,

the total number of containers discharged by all vessels calling during May is aggregated. Based on these container volumes, the corresponding number of trucks required for pick-up operations is derived.

As a result, predictions are obtained for the number of trucks expected to reach the terminal within the FIs of the vessels calling during May 2023. The one-month planning horizon is therefore represented by a matrix in which each row corresponds to the predicted number of trucks assigned to container pick-up operations for each day of the month. For days falling outside the FI of a specific vessel, the associated number of trucks is set equal to zero.

In port terminal operations, specific yard areas are typically assigned to handle import containers from different services. This allocation is generally determined by operational considerations such as container volumes, vessel size, or service characteristics. In cases where multiple services allocate import containers to the same yard area, the predicted truck arrivals for that area are computed by aggregating the daily contributions associated with each relevant service.

It should be noted that, although the structural layout of the PSA GP terminal is adopted as reference for the areas considered in this study, minor modifications have been introduced for confidentiality reasons and to facilitate the presentation of results. These adjustments concern the allocation of services to yard areas and the corresponding percentage shares of containers assigned to each service.

Within the previously described optimization framework, two key parameters are the daily and hourly capacity thresholds for truck arrivals in a given area, denoted by T_{max} and Tr_{max} , respectively. The evaluation of these parameters depends on the operational configuration of the area under consideration. As an illustrative example, an area operated by Rubber Tyred Gantry (RTG) cranes, where containers are transferred from their storage slots to trucks, is considered.

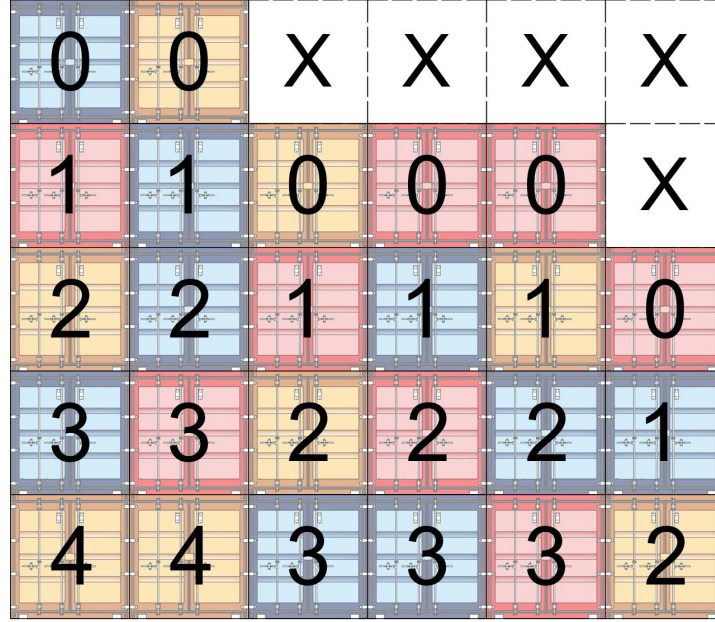


Figure 7.1: Container positioning and repositioning moves

Fig. 7.1 illustrates the front view of a container bay. The numbers indicated on each container denote the repositioning moves required by the crane to retrieve that container from its storage slot and load it onto a truck. Slots marked with an 'X' represent positions where containers cannot be stored. These include four slots reserved to facilitate repositioning operations and one additional slot reserved for safety purposes.

For example, in order to retrieve the container located in the bottom-left position, four preliminary moves are required to remove the containers stacked above it, followed by one additional move to extract the target container. Therefore, assuming the container arrangement shown in Fig. 7.1, the average number of repositioning moves (RM) performed by the RTG crane to retrieve a single container is computed as follows:

$$\overline{RM} = \frac{1}{t * r} \sum_{i=1}^t \sum_{j=1}^r RM_{ij} = \frac{41}{25} \approx 1.7 \quad (7.18)$$

where t is the number of tiers, r is the number of rows, and RM_{ij} is the number of repositioning moves necessary at slot i, j . Then, the number of RTG moves necessary to serve one truck is $(1.7 + 1)$ [moves/truck], where 1 is added for the movement of the desired container.

Furthermore, assuming that the average operational productivity of an RTG during the truck

service cycle is equal to 22 moves per hour, the number of trucks that can be served within one hour by a single RTG can be estimated as

$$\overline{Tr} = \frac{22}{(1.7 + 1)} \approx 8.$$

Finally, under the assumption that only one RTG operates in the considered area, the parameters adopted in the optimization framework are set equal to $Tr_{max} = 8$ and $T_{max} = 128$.

These values are derived from the assumed configuration of handling resources available in the selected area, which reflects the operational characteristics of certain areas of PSA GP. However, such parameters can be readily recalculated under different operational conditions, thereby allowing the optimization framework to be tailored to the specific handling processes and resource configurations of each terminal area.

This section presents the outcomes of the optimization approach. For presentation purposes, the analysis is focused on a single area, namely area B02. This area was selected because of its significance, as it accommodates containers from one of the largest services and is therefore exposed to a higher risk of congestion.

The results of the first phase of the optimization problem are shown in Figs. 7.2a and 7.2b where it is shown that the first optimization phase effectively redistributes trucks to ensure that the daily limit is not exceeded.

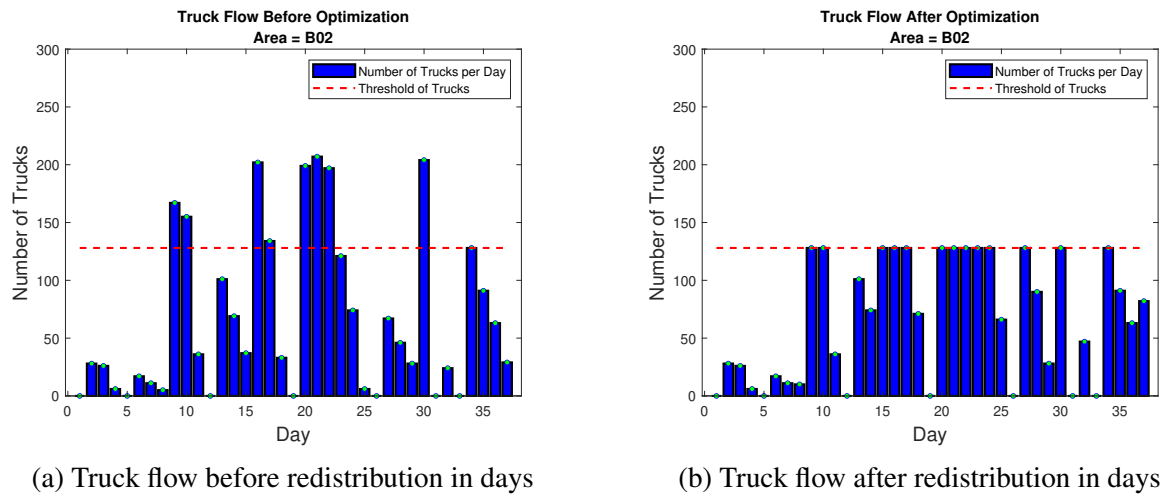


Figure 7.2: Daily truck flow before and after redistribution

Following the successful redistribution of truck flows across days in the first phase, the second phase focuses on optimizing truck arrivals within each hour of the day, with the objective of preventing congestion in specific terminal areas caused by excessive hourly peaks. At this stage,

the optimized daily truck distribution obtained from the first phase for the same area is adopted as input and further refined by considering a specific day within the planning horizon.

Day 13 is selected as a representative case, to which Figs. 7.3a and 7.3b refer. The results clearly indicate that the proposed approach effectively redistributes truck arrivals so that hourly peaks exceeding the predefined threshold are eliminated. Moreover, the number of adjustments introduced by the optimization remains limited, thereby ensuring minimal disruption to the original operational pattern.

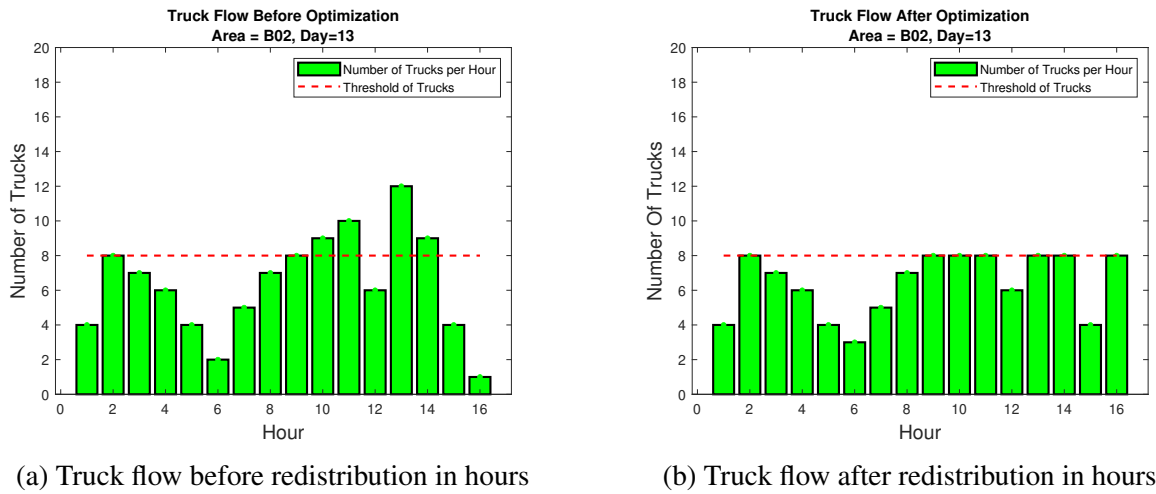
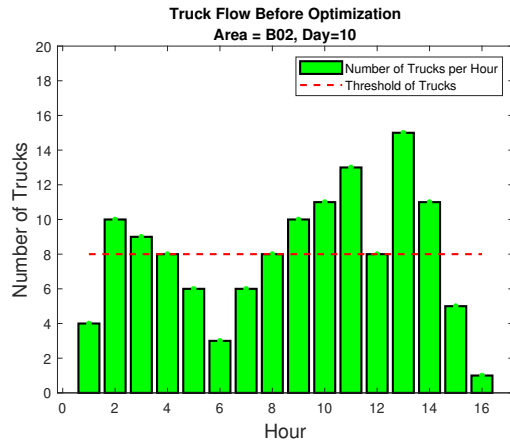


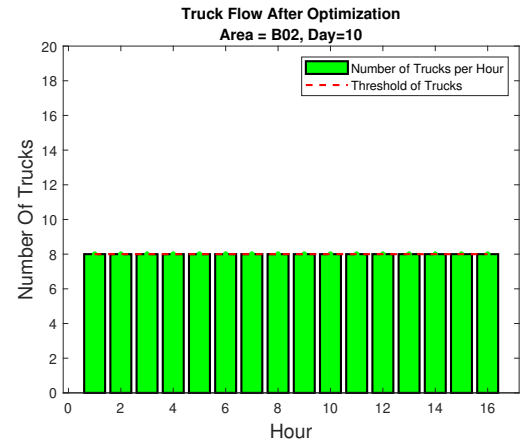
Figure 7.3: Hourly truck flow before and after redistribution (Day 13)

A more representative case is provided by day 10, for which the results of the second optimization phase are shown in Figs. 7.4a and 7.4b. As illustrated in the figures, truck arrivals are effectively redistributed across the hourly intervals, ensuring that the predefined threshold of 8 trucks per hour is not exceeded.

It should be emphasized that, prior to the first optimization phase, day 10 was characterized by a very high predicted number of truck arrivals, significantly exceeding the daily capacity limit. Consequently, without the preliminary redistribution across days, it would not have been possible to obtain a feasible solution to the second optimization problem without hourly peaks. The initial redistribution of truck arrivals across days, whenever feasible, therefore plays a crucial role in enabling the second optimization phase to achieve a solution in which no hourly threshold violations occur.



(a) Truck Flow Before Redistribution in Hours



(b) Truck flow after redistribution in hours

Figure 7.4: Hourly truck flow before and after redistribution

Overall, these findings underscore the importance of a two-phase optimization approach in effectively managing truck flows and mitigating congestion risks within specific areas.

7.2 Implementation of emission control strategies

The second motivation for employing the predictions developed in Chapter 6 is the estimation of emissions generated within the terminal area.

Emission calculations are performed according to the COPERT methodology described in Section 4.2. For each considered pollutant, the corresponding emission levels are estimated based on the predicted truck flows obtained from the forecasting framework.

After computing the emissions associated with a specific pollutant, two alternative mitigation schemes are developed. The first is based on an algorithmic approach, while the second relies on an optimization model conceptually similar to the truck redistribution framework previously introduced for congestion reduction. In this case, however, the objective is not to smooth traffic peaks in terms of truck volumes, but rather to redistribute truck arrivals in order to reduce peaks in emissions generated within the terminal area.

7.2.1 Case study

Based on the definition provided in Section 4.2, the total emissions are calculated as shown in Equation 4.4.

In this case, the emission estimation is based on the predicted number of trucks arriving at the terminal to deliver export containers. This prediction framework is described in Section 6.1. For the specific case study, the cumulative predictions for the month of May 2023 are considered. The input to the emission estimation model corresponds to the truck flow illustrated in Fig. 7.5.

For completeness, a brief operational overview is recalled here. After gate clearance, trucks proceed to the designated export area to discharge containers, provided that no congestion is present. In cases where the assigned area is congested due to high truck volumes, vehicles are redirected to a buffer zone where they wait until capacity becomes available. Once the area is ready to receive additional trucks, drivers are informed through a dedicated mobile application or SMS notification. The containers are then discharged, and the trucks exit the terminal.

It should be emphasized that the waiting time in the buffer area is not sufficient for engine cooling. Similarly, although truck engines are switched off during the unloading operation performed by the RS, statistical data extracted from the PSA GP database indicate that the average time spent within the terminal is approximately 25 minutes, which is not long enough to allow the engine to cool down completely.

Therefore, cold-start emissions (E_{cold}) are not considered in the emission estimation process.

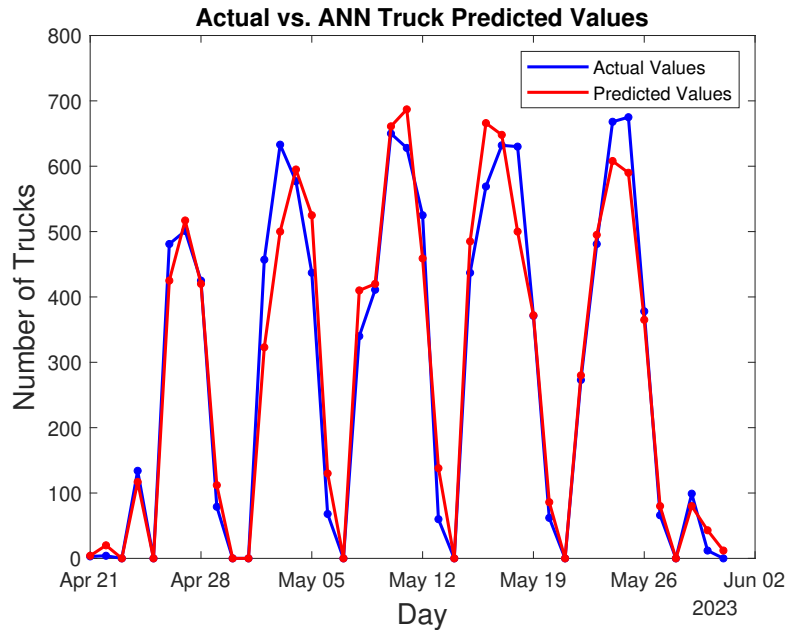


Figure 7.5: Actual vs. ANN predicted truck arrival curve for the period taken into consideration

7.2.2 Factors for emission factor estimation

In this subsection, the main inputs required to compute the hot emission coefficient, as defined in Equation 4.6, are described. For the calculation of hot emissions, the primary factors considered and the specific assumptions adopted are outlined below.

- In place of $M_{k,r}$, it is considered the overall average distance covered by trucks inside the port area which results to be approximately 5 km.
- The primary objective of this study is to introduce a model for estimating and redistributing emissions, so only one pollutant, namely CO, is considered. Of course, for other pollutants the same method can be applied.
- The trucks arriving at PSA GP are articulated 34-40 tonnes Diesel Trucks.
- Since most of the trucks arriving to the terminal are equipped with Euro V engine technology, only this technology is considered in the computation of the emissions.
- The emission function parameters provided for Euro V vehicles, depend on their emission control concept. Since, for European Member States, it is estimated that approximately 75% of Euro V heavy-duty vehicles are equipped with Selective Catalytic Reduction (SCR) and the rest are equipped with Exhaust Gas Recirculation (EGR), an assumption that we make is that all the trucks arriving to the terminal are equipped with SCR.
- In terms of trip specifics, a road slope of 0 is chosen, indicating a perfectly level or flat road.
- According to the Guidebook, by default, a factor of 50% is considered for a load of heavy-duty vehicles, so the emission coefficient is computed using a load factor equal to 0.5.
- Regarding speed, we account for an average speed of 30 km/h for trucks operating within the terminal.

In Table 7.1, the essential factors for the emission calculation relevant to the case study are summarized. Taking into account these factors and some specific assumptions, the emission coefficient has been set equal to $EF = 2.4 \text{ g/km}$.

It is important to note that the proposed methodology is effective for all pollutants for which, according to the guidebook, the described tier 3 methodology can be used. However, for other pollutants, the solution is limited by the possibility of obtaining the necessary data required by the other tiers, such as the amount of fuel consumed. The same discussion applies to other truck technologies or types; having accurate information about all the different vehicle technologies and types may increase the accuracy of the work. Meanwhile, evaporative emissions, though

Table 7.1: Main factors for EF_{hot} calculation

Characteristic	Value
Category	Heavy Duty Trucks
Fuel	Diesel
Segment (Truck Type)	Articulated 34 - 40 t
Euro Standard	Euro V
Technology	SCR
Pollutant	CO
Road Slope	0
Load	0.5
Avg. Speed	30 [km/h]

important, are not as crucial for calculating the amount of each pollutant; instead, they are significant for Volatile Organic Compounds.

The dataset being analyzed consists of the outcomes generated by the predictive methodology, which specifically pertain to the predicted number of trucks for each time interval $[YO, YC]_s$ corresponding to every scheduled ship s arriving in May 2023. The terminal yard is prepared to accept containers via trucks for a particular ship at least 7 days before its scheduled arrival. Consequently, the earliest value of YO_{min} occurs in the month of April.

The total emissions are computed as the product of the predicted number of trucks, the distance traveled within the terminal, and the corresponding emission factor. The following two subsections present alternative methodologies for the redistribution of truck arrivals with the objective of mitigating emissions in cases where the estimated levels exceed predefined allowable thresholds.

7.2.3 Redistribution algorithm

The proposed redistribution algorithm incorporates the operational constraints characterizing terminal activities. For instance, trucks delivering export containers for vessel loading are allowed to access the terminal only within predefined time windows. In addition, practical considerations are taken into account, as trucking companies are generally reluctant to substantially modify their schedules. Therefore, the redistribution strategy is designed to introduce only limited adjustments to the original operational plan. The algorithm is developed to redistribute truck-related emissions over a specified time horizon with the objective of eliminating emission peaks. In particular, it aims to ensure that the estimated emission levels remain below a predefined threshold.

Algorithm 3 Emission redistribution algorithm

```
1: For each  $s$ -th ship and days  $d$  in the interval  $[YO_{min}, YC_{max}]$ 
2:  $N_d \leftarrow \sum_{d \in [YO, YC]_s} Nr(d)_{Trc}$ 
3:  $E_d \leftarrow Nr(d)_{Trc} * D * EF$ 
4: for  $d \in [YO_{min}, YC_{max}]$  do
5:   if  $E_d > E_{max}$  then
6:      $\underline{d}_{excess} \leftarrow d$ 
7:      $\underline{E}_{diff} \leftarrow E_{max} - E_d$ 
8:      $\underline{T}_{diff} \leftarrow \frac{\underline{E}_{diff}}{(M \cdot EF)}$ 
9:   end if
10:  for  $s \leftarrow 1$  to  $N$  do
11:    if  $\underline{d}_{excess} \in [YO, YC]_s$  then
12:       $\underline{E}_{portion} = \frac{\underline{E}_{d_{excess}}}{E_d} \cdot \underline{E}_{diff}$ 
13:       $\underline{T}_{portion} = \frac{\underline{E}_{portion}}{(D \cdot EF)}$ 
14:    end if
15:  end for
16: end for
17: ▷ Start redistribution process
18: for  $s \leftarrow 1$  to  $N$  do
19:    $R \leftarrow [YO, YC]_s - [d_{excess}, d_{holiday}]$ 
20:   for  $d_{remain} \in R$  do
21:      $dist \leftarrow ||d_{remain} - d_{excess}||$ 
22:      $w \leftarrow \frac{1}{dist}$ 
23:      $Nr(d)_{Trc} \leftarrow \sum_{d \in d_{remain}} T_{portion} \cdot w$ 
24:   end for
25: end for
```

Then, the redistribution algorithm works in the following way:

- For each ship predicted to arrive in the month M_j ($j = 1 \dots 12$) the trucks predicted to arrive in the interval $[YO, YC]_s$ for each ship s are evaluated
- $[YO, YC]_s$ refers to the time interval between yard opening (YO) and yard closing (YC) dates, indicating the time interval during which trucks are permitted to arrive for exporting containers for each ship s . Trucks carrying containers to be exported with the specified ship must respect the constraint that they cannot arrive before the YO date or after the specified YC date. These constraints are considered in the redistribution algorithm
- YO_{min} and YC_{max} refer to the minimum and maximum date of all intervals, respectively

- E_{max} refers to the emission threshold that should not be exceeded
- $Nr(d)_{Trc}$ refers to the number of trucks predicted in each day d of each $[YO, YC]_s$
- D is the average distance covered by the trucks within the terminal
- EF is the emission factor
- N_d and E_d refer to the total number of trucks and emissions predicted on day d of $[YO_{min}, YC_{max}]$, with contributions from trucks for each ship s
- d_{excess} contains the days that exceed the threshold
- E_{diff} and T_{diff} refer to the differences in emissions and trucks for the days that exceed the threshold
- $E_{portion}$ and $T_{portion}$ refer to the portion of emissions and trucks in a specific d_{excess} that needs to be distributed in the interval of each ship s that contains d_{excess}
- R refers to the interval of remaining days after removing holidays and days with excess emissions from $[YO, YC]_s$
- $dist$ refers to the distance in time between each day with exceeding emissions and the remaining days.
- w refers to the weight given to each remaining day in order to distribute the trucks from d_{excess} to the closest remaining days.

The results of the proposed redistribution algorithm are presented below.

Figs. 7.6a and 7.6b illustrate the estimated CO emissions derived from the predicted truck arrivals. In the first application of the algorithm, a threshold value of $E_{max} = 6800$ was adopted. As shown in Fig. 7.6b, the redistribution process succeeds in maintaining emissions below the selected threshold in almost all cases, while respecting the operational constraints embedded in the algorithm. The proportion of trucks subject to redistribution corresponds to approximately 4% of the total truck arrivals.

It should be emphasized that selecting a lower emission threshold may lead to situations in which no feasible solution exists, provided that all operational constraints are strictly enforced. In particular, feasibility may be compromised when truck arrivals must remain within the admissible time windows $[YO, YC]$ and when deviations from the predicted arrival day are required to remain minimal. In such cases, an iterative approach can be adopted, starting from a relatively high value of E_{max} and progressively decreasing it at each iteration.

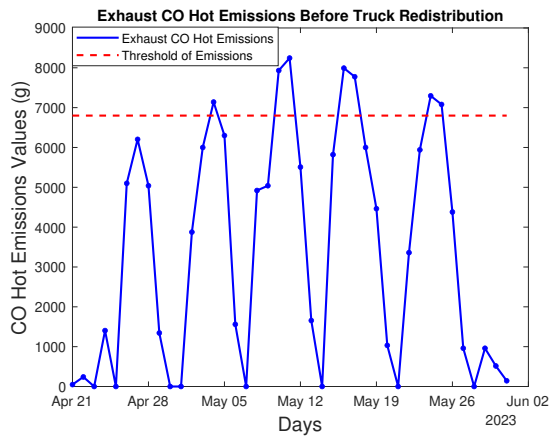
For $E_{max} = 6800$, a satisfactory solution is obtained in the first iteration of the algorithm. However, when smaller threshold values are imposed, multiple iterations may be necessary in order

to achieve an acceptable redistribution. Although the results presented refer to a specific case study and focus on a single pollutant (CO), the proposed methodology can be applied to different operational contexts and extended to any pollutant, provided that the appropriate emission factor parameters are defined.

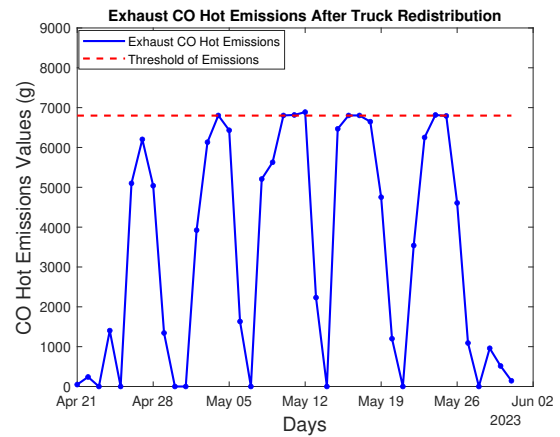
It is nevertheless important to underline that any redistribution solution must be considered in light of its practical acceptability. The willingness of road transport operators to accommodate schedule modifications decreases as the proportion of redistributed trips increases, which may limit the practical implementation of more aggressive emission reduction strategies.

As an illustrative example, the results obtained when a lower threshold of $E_{max} = 5800$ is adopted are shown in Figs. 7.7–7.8.

Fig. 7.7b reports the outcome after the first iteration of the algorithm. Although emissions are reduced on days where the threshold was initially exceeded, increases are observed on other days due to the redistribution mechanism. Subsequent iterations lead to improved results, as illustrated in Fig. 7.8b, which presents the solution obtained after the sixth iteration (with no further improvement achieved in the seventh iteration). In this case, approximately 10% of the total trucks are redistributed.

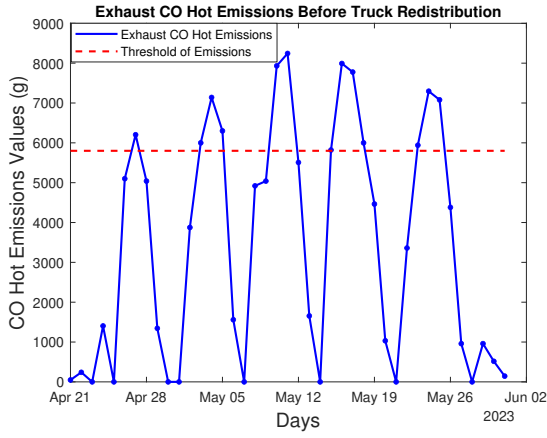


(a) Exhaust CO Hot Emissions before truck redistribution ($E_{max} = 6800$)

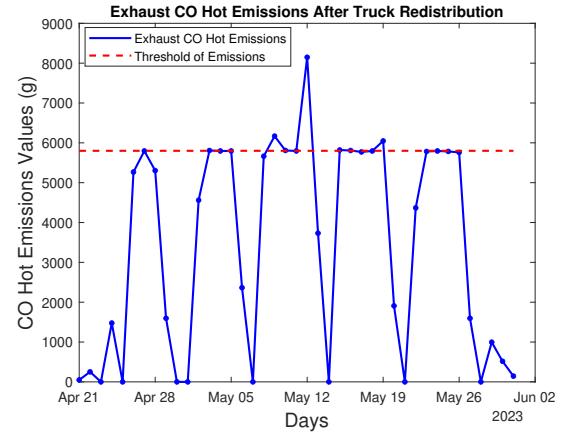


(b) Exhaust CO Hot Emissions after truck redistribution ($E_{max} = 6800$)

Figure 7.6: Exhaust CO Hot Emissions before and after truck redistribution

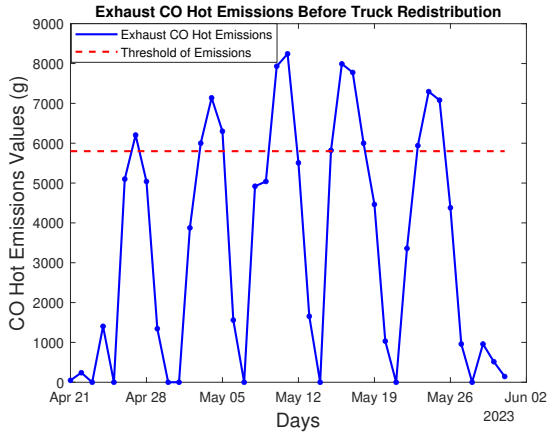


(a) Before truck redistribution ($E_{max} = 5800$)

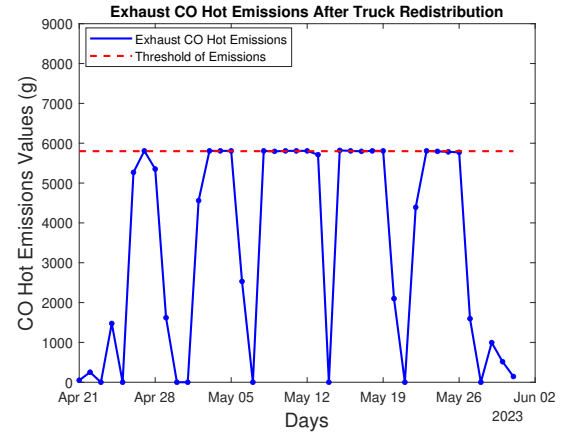


(b) After first iteration ($E_{max} = 5800$)

Figure 7.7: Exhaust CO Hot Emissions before and after the first iteration of the redistribution algorithm



(a) Before truck redistribution ($E_{max} = 5800$)



(b) After sixth iteration ($E_{max} = 5800$)

Figure 7.8: Exhaust CO Hot Emissions before and after the sixth iteration of the redistribution algorithm

Although the results presented refer to a specific case study and focus on a single pollutant, the proposed methodology is general in nature and can be applied to different operational contexts and to any pollutant. The only requirement is the appropriate selection of the parameters needed for the computation of the corresponding emission factors.

It should nevertheless be emphasized that, despite the substantial reduction of emission peaks achieved by the algorithm, the practical implementation of any redistribution solution depends on

the acceptance of the road transport operators involved. Such acceptance is expected to decrease as the proportion of redistributed trips increases, potentially limiting the applicability of more extensive emission mitigation strategies.

7.2.4 Emission optimization problem

The second approach for emission mitigation is based on an optimization model for truck redistribution. In this case, the problem is formulated as the redistribution of truck arrivals across the days of the planning horizon, with the objective of minimizing the amount of emissions exceeding the predefined threshold value E_{max} .

The decision variables included in the optimization model are defined as follows:

- $n_{s,t}^+, s = 1, \dots, S, t = 1, \dots, T, n_{s,t}^+ \in \mathbb{R}^+$, is the number of trucks added to the normal planning in day t for ship s .
- $n_{s,t}^-, s = 1, \dots, S, t = 1, \dots, T, n_{s,t}^- \in \mathbb{R}^+$ is the number of trucks deducted to the normal planning in day t for ship s .
- $n_{s,t}, s = 1, \dots, S, t = 1, \dots, T, n_{s,t} \in \mathbb{R}^+$ is the resulting number of trucks for ship s in day t .

The statement of the optimization problem follows.

Problem 5

$$\min \sum_{t=1}^T \max \left(\sum_{s=1}^S n_{s,t} - N_{max}, 0 \right) \quad (7.19)$$

subject to

$$n_{s,t} = d_{s,t} + n_{s,t}^+ - n_{s,t}^- \quad s = 1, \dots, S \quad t = 1, \dots, T \quad (7.20)$$

$$\sum_{t=1}^T n_{s,t} = \sum_{t=1}^T d_{s,t} \quad s = 1, \dots, S \quad (7.21)$$

$$\sum_{s=1}^S n_{s,t}^- \leq \max \left(\sum_{s=1}^S d_{s,t} - N_{max}, 0 \right) \quad t = 1, \dots, T \quad (7.22)$$

$$n_{s,t}^+ = n_{s,t}^- = 0 \quad s = 1, \dots, S, t \in \mathcal{T}_{off} \quad (7.23)$$

$$n_{s,t}^+ = n_{s,t}^- = 0 \quad s = 1, \dots, S, t < YO_s \quad (7.24)$$

$$n_{s,t}^+ = n_{s,t}^- = 0 \quad s = 1, \dots, S, t > YC_s \quad (7.25)$$

The considered cost function to be minimized is the total number of trucks exceeding the threshold N_{max} . Constraints (7.20) define the number of trucks reaching the terminal for ship s at day t . Constraints (7.21) ensure the conservation of the total number of trucks for each ship after redistribution. Constraints (7.22) impose that the maximum decrease in the number of trucks reaching the terminal at day t is equal to the number of trucks exceeding the threshold without redistribution. These constraints are useful to impose that the minimum changes are done to the predicted arrivals. Finally, constraints (7.23), (7.24) and (7.25) ensure that trucks are not redistributed to days in which the terminal is closed or outside the time interval $[YO_s, YC_s]$ for each ship s .

Problem 5 is further simplified by linearizing the cost function, as in the following formulation.

Problem 6

$$\min \sum_{t=1}^T C_t \quad (7.26)$$

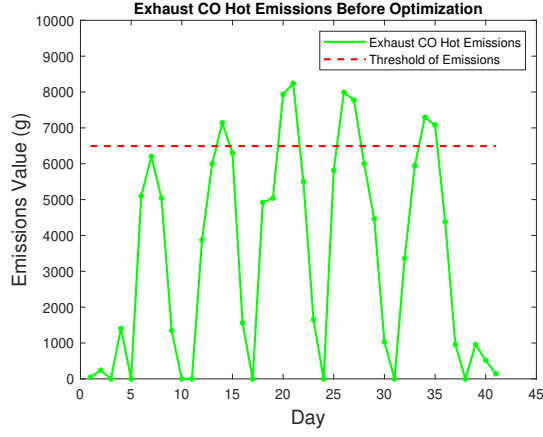
subject to (7.20), (7.21), (7.22), (7.23), (7.24), (7.25), and

$$C_t \geq \sum_{s=1}^S n_{s,t} - N_{max} \quad t = 1, \dots, T \quad (7.27)$$

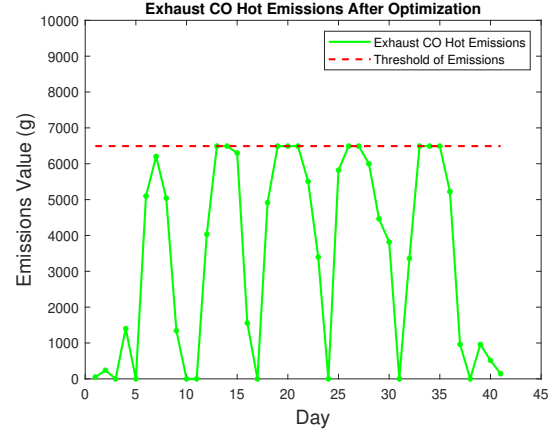
$$C_t \geq 0 \quad t = 1, \dots, T \quad (7.28)$$

The results of the truck redistribution obtained by solving Problem 6 are here reported. Fig. 7.9a displays the CO emissions resulting from the predicted number of trucks arriving during the considered time period.

At first, a threshold emission level of approximately 6500, specifically $E_{max} = 6492$, has been selected. As shown in Fig. 7.9b, an optimal solution is found for the optimization problem, and following the redistribution of trucks while respecting all constraints, the emission levels do not exceed the threshold E_{max} on all days. The redistributed trucks account for 5.674% of the total number of trucks.

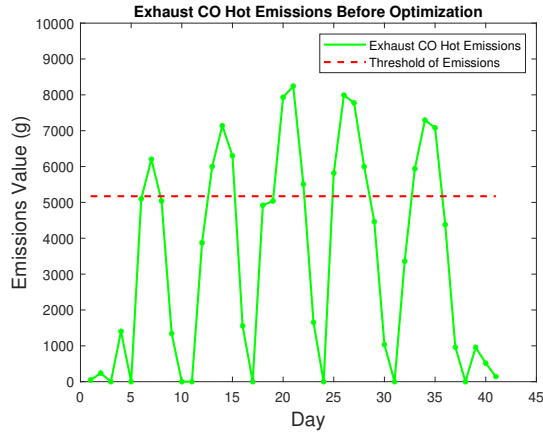


(a) Exhaust CO Hot Emissions before truck redistribution ($E_{max} = 6492$)

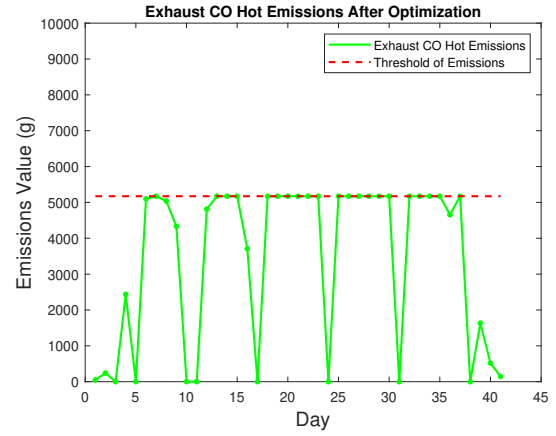


(b) Exhaust CO Hot Emissions after truck redistribution ($E_{max} = 6492$)

Figure 7.9: Exhaust CO Hot Emissions before and after optimization-based truck redistribution ($E_{max} = 6492$)



(a) Exhaust CO Hot Emissions before truck redistribution ($E_{max} = 5172$)



(b) Exhaust CO Hot Emissions after truck redistribution ($E_{max} = 5172$)

Figure 7.10: Exhaust CO Hot Emissions before and after optimization-based truck redistribution ($E_{max} = 5172$)

Meanwhile, Figs. 7.10a and 7.10b show the predicted emissions, with a lower threshold, and the results after optimization. The threshold E_{max} in this instance is set to 5172, the lowest value for which an optimal solution is attainable (for lower thresholds, achieving an optimal solution that satisfies all constraints can become unfeasible). Approximately 16.156% of the total trucks have

been redistributed. The figures illustrate a notable reduction in emission peaks, indicating the feasibility of the problem.

It can be observed that the optimization-based formulation provides a more structured and systematic approach to addressing the emission mitigation problem. In addition, it is capable of identifying an optimal solution within a very short computational time. Both methodologies have been implemented in the MATLAB environment.

Related publications. The work presented in this chapter has been published in the following papers: [117], [118] and [119].

Chapter 8

Rail Operation Optimization

As previously stated in this thesis, one of the key subsystems under investigation is the rail subsystem, which represents a fundamental component of the PSA Genova Pra' (PSA GP) terminal. The research activities conducted in this context focus on two main aspects: the optimization of the rail discharge process and the evaluation of the overall capacity of the rail yard.

An overview of the adopted methodology is provided below, while a detailed description is presented in the following sections.

The framework illustrated in Fig. 8.1 describes the approach developed to address the first research objective related to the rail subsystem, namely the optimization of operational processes within the rail yard. The procedure begins with the extraction of relevant operational data from the company's data warehouse through structured SQL queries, specifically targeting train-related information. Based on the collected data, and with the objective of improving the efficiency of the train discharge process, a scheduling optimization problem is formulated.

The scheduling problem is addressed using two complementary solution approaches. First, a mathematical programming formulation is developed to obtain optimal discharge schedules. However, due to its computational complexity, this exact approach is applicable only to small problem instances. For larger and more realistic instances, a heuristic solution model is therefore proposed. The heuristic approach is capable of generating near-optimal discharge schedules within reasonable computational times, making it suitable for practical application.

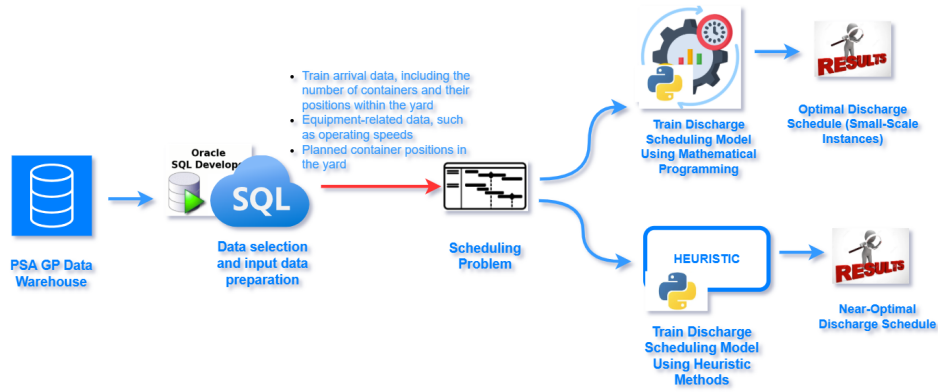


Figure 8.1: Conceptual framework of the research for rail discharge optimization

8.1 Optimized scheduling of train discharge

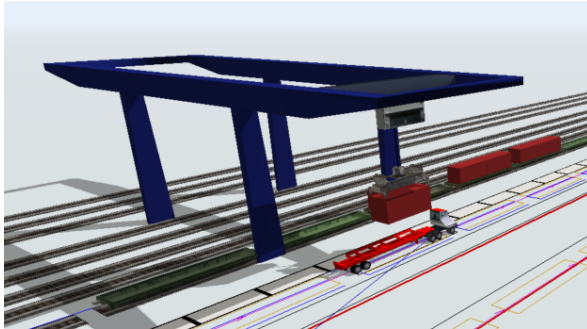
The rail area of the terminal consists of two rail parks, each equipped with eight tracks, thereby allowing the accommodation of up to eight trains simultaneously in each park.

The focus of this thesis is exclusively on the train discharging process. This operation is relatively straightforward, as trains arrive with a fully known and predefined composition. Detailed information is available for each container, including its characteristics and its position on the corresponding wagon. These predefined initial conditions constitute essential inputs to the system and simplify the modeling of the process.

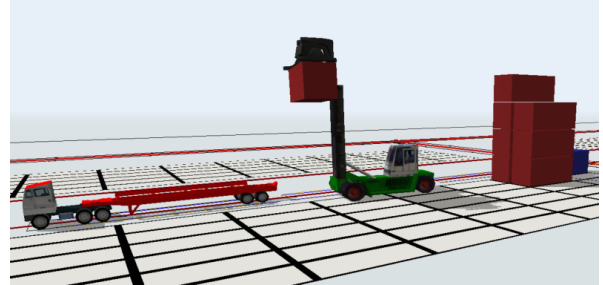
The discharging operation begins with Rail Mounted Gantry (RMG) cranes lifting containers from the train wagons and placing them onto Prime Movers (PMs), as illustrated in Fig. 8.2a. PMs as mentioned before, are internal transport vehicles operating within the terminal, responsible for transferring containers from the rail area to designated storage locations in the yard. Once the containers reach the assigned yard areas, Reach Stackers (RS), as shown in Fig. 8.2b, unload the containers from the PMs and position them within the corresponding yard blocks.

8.1.1 Current way of operating

Each RMG operating within a rail park is capable of serving up to eight tracks simultaneously, corresponding to a maximum of eight trains if all tracks are occupied. Since each rail park is equipped with two RMGs, this configuration allows significant operational flexibility. However, under the current operational conditions, it is typically observed that only one RMG is active during most shifts, and it is assigned to handle a single train at a time. The crane proceeds to the next train only after completing all discharging operations on the current one.



(a) RMG discharging a container from the train and placing it onto a PM



(b) RS discharging a container from the PM and placing it in the container stack.

Figure 8.2: Main phases of the rail container discharging process: transfer from train to PM and from PM to yard stack

Although the process is sequential in nature, certain operational strategies are implemented in order to improve efficiency in terms of both time and cost. These strategies include:

- **Prioritization by container type.** Discharging operations generally begin with containers of the same type, either empty or full, since these are allocated to different yard areas. This approach reduces the travel distance required for the RS, which transfers containers from the PM to the designated storage blocks. Given that the RS operates at relatively low speeds, minimizing unnecessary movements between yard areas contributes to improved operational performance.
- **Prioritization by container size.** When handling containers of the same type (empty or full), operations are typically organized by size category, either 20-foot or 40-foot containers. This sequencing reduces the frequency of spreader adjustments on the RMG, as changing the spreader configuration requires additional time and may decrease overall productivity.

The objective of this research is to enhance the train discharging process, which is currently not optimized. In particular, the aim is to formulate and solve a scheduling problem that organizes container discharge operations in an efficient manner, while respecting the relevant operational constraints and ensuring effective utilization of the available cranes and handling resources. Such improvements are consistent with the terminal's operational and business objectives.

This study constitutes an initial step toward the broader goal of optimizing both train discharging and loading processes. It establishes a structured methodological foundation that may support future developments, including the coordinated operation of multiple RMGs for both discharge and loading activities.

8.1.2 Exact solution

The first approach to the problem consists of formulating it as a mathematical programming model and deriving an exact optimal solution. However, this approach is computationally tractable only for relatively small problem instances, as will be shown through the problem formulation and the results presented in the following subsections.

8.1.2.1 Flexible flow-shop scheduling problem definition

To translate the real-world operational system into a mathematical framework, the process is modeled according to the logic of the Flexible Flow Shop Scheduling Problem (FFSSP), which is widely recognized for its potential to enhance efficiency in terms of time, cost, and energy consumption [120].

As described in the Methodology chapter, the FFSSP extends the classical flow shop scheduling problem by introducing machine flexibility. In this formulation, each operation can be processed by at least one machine selected from a subset of eligible machines. The solution of a flexible flow shop problem involves two interconnected components: the routing sub-problem, which assigns each operation to a suitable machine among the available alternatives, and the scheduling sub-problem, which determines the sequence of operations on the selected machines in order to optimize a predefined objective function.

In the FFSSP framework, a set of J jobs is processed through S consecutive stages, where each stage comprises a subset of M parallel machines. Consequently, each job j may be assigned to any eligible machine at each stage, thus providing flexibility in resource allocation and improving overall utilization [121], [122].

Within the rail discharge process, the handling vehicles are modeled as machine sets M , specifically:

- M_C : the set of RMGs, by which every container must be processed;
- M_P : the set of PMs, among which each container can be assigned;
- M_R : the set of RSs, by which each container must also be processed.

Since each container is required to pass sequentially through the RMG, PM, and RS stages, while retaining flexibility in the selection of PMs and RSs, the problem naturally fits within the FFSSP framework.

In the present formulation, a single RMG is considered. Nevertheless, the model can be extended to include multiple RMGs by introducing additional constraints to represent spatial and operational interactions between cranes.

Each container is modeled as a job j . Multiple train tracks are available for arriving trains, each characterized by a known but variable number of containers. The time spent by a container on a specific handling vehicle is modeled as the processing time of the corresponding job on the associated machine. More precisely, for the RMG, the processing time corresponds to the transfer of the container from the train to the PM; for the PM, it represents the transportation time to the designated yard area and the return trip; and for the RS, it corresponds to the placement of the container from the PM into the yard block.

The time required for a vehicle to transition from handling one container to the next is modeled as setup time. For the RMG, setup time includes:

- the horizontal displacement required when the subsequent container is located on a different wagon along the train;
- the adjustment of the spreader when containers of different sizes are handled, in cases where this adjustment is not completed during crane movement;
- in scenarios involving multiple trains, the trolley movement required to transition between containers located on different trains.

For the RS, setup time is defined as the travel time required to move from the stacking position of the previous container to the location where the subsequent container must be placed.

Finally, the total horizontal distance traveled by a PM depends on the discharge sequence adopted, whereas the total vertical hoisting movement of the RMG is independent of the sequencing decisions. For this reason, only the sequence-dependent horizontal movements are considered within the optimization model.

8.1.2.2 FFSSP formulation

As mentioned above, the FFSSP consists of J jobs to be processed on M machines, divided into different consecutive stages, according to specific production constraints. In the proposed system, machines are divided into three different sets: M_C, M_P, M_R , which represent respectively the set of RMG cranes, PMs, and RSs of the port terminal. The sets of machines is denoted as $\mathcal{M} = \{M_C, M_P, M_R\}$.

The decision variables of the problem are:

- $s_{j,m}$: start time of container j on machine m ;
- $c_{j,m}$: completion time of container j on machine m ;

- $w_{j,m}$: waiting time of container j on machine m ;
- $\delta_{i,j,m}$: binary variable to manage precedence constraints, equal to 1 if job i precedes job j on machine m , and 0 otherwise;
- $x_{i,j,m}$: binary variable to manage immediate precedence constraints, equal to 1 if job i immediately precedes job j on machine m , and 0 otherwise;
- $\gamma_{j,m}$: binary variable equal to 1 if job j is assigned to machine m , and 0 otherwise;
- C^{max} : value of the makespan.

The parameters of the problem are:

- $u_{i,j,m}$: setup time required for machine m to move from container i to container j . This variable models the distance between two containers (similar to an origin-destination matrix);
- $v_{i,j,m}$: setup time required for the spreader adjustment when transitioning from container i to container j . It is 0 if both containers are of the same size; otherwise, it corresponds to the time needed to adjust the spreader's opening for the RMG when switching between a 20-foot and a 40-foot container, or vice versa.
- $p_{j,m}$: processing time of container j on machine m ;

The objective function of the problem is written as:

$$\min C^{max} + \epsilon \sum_{i=1}^J \sum_{j=1}^J \sum_{m \in \mathcal{M}} (s_{i,j,m} + c_{i,j,m} + w_{i,j,m}) \quad (8.1)$$

where ϵ is a term used to nullify start, completion, and waiting times that are not selected by the optimal solution, added to ensure consistency among the constraints of the problem, as presented below. This cost function differs from the empirical solution used by the terminal, where horizontal movement is minimized. However, the two objectives are strongly correlated, as minimizing horizontal movement generally leads to the optimization of the total discharging time.

The system constraints are:

$$c_{i,m} \geq s_{i,m} + p_{i,m} + (u_{i,j,m} + v_{i,j,m})x_{i,j,m} + (1 - \gamma_{i,m})\mathcal{B} \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2a)$$

$$\sum_{m \in M_y} \gamma_{j,m} = 1 \quad M_y = \{M_C, M_P, M_R\} \quad \forall j = 1, \dots, J \quad (8.2b)$$

$$s_{j,m'} \geq c_{j,m} - (1 - \gamma_{j,m})\mathcal{B} \quad \forall j = 1, \dots, J, \forall m \in M_y, m' \in M_{y+1}, \quad M_y = \{M_C, M_P, M_R\} \quad (8.2c)$$

$$\sum_{i=1}^J x_{i,j,m} \leq 1 \quad \forall j = 1, \dots, J, \quad \forall m = 1, \dots, M \quad (8.2d)$$

$$\sum_{j=1}^J x_{i,j,m} \leq 1 \quad \forall i = 1, \dots, J, \quad \forall m = 1, \dots, M \quad (8.2e)$$

$$\sum_{j=1}^J \sum_{i=1}^J \sum_{m \in M_y} x_{i,j,m} \leq J - \#M_y \quad \forall i, j = 1, \dots, J, i \neq j, M_y = \{M_C, M_P, M_R\} \quad (8.2f)$$

$$x_{i,j,m} + x_{j,i,m} = 1 \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2g)$$

$$\delta_{i,j,m} + \delta_{j,i,m} = 1 \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2h)$$

$$x_{i,j,m} \leq \delta_{i,j,m} \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2i)$$

$$\delta_{i,j,m} \leq \gamma_{i,m} \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2j)$$

$$s_{j,m} \geq c_{i,m} + w_{i,m} - (1 - \delta_{i,j,m})\mathcal{B} \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2k)$$

$$s_{i,m} \geq c_{j,m} + w_{j,m} - (1 - \delta_{j,i,m})\mathcal{B} \quad \forall i, j = 1, \dots, J, i \neq j, \forall m \in \mathcal{M} \quad (8.2l)$$

$$w_{j,m} \geq \sum_{m' \in M_{y+1}} s_{j,m'} - c_{j,m} - (1 - \gamma_{j,m})\mathcal{B} \quad \forall j = 1, \dots, J, m \in M_y \quad (8.2m)$$

$$C^{max} \geq c_{j,m} \quad \forall j = 1, \dots, J, \forall m \in M_R \quad (8.2n)$$

Constraint (8.2a) expresses the relationship between completion time and start time, also considering the setup times required to transition from one container to another and for the potential opening of the spreader. Since the spreader can open while the RMG crane is moving, the calculation of setup v depends on the distance between the two containers, hence on u , avoiding the setup for the first scheduled container (assuming the RMG crane is already there; otherwise,

a fictitious container is created representing the crane's starting point). The total setup time is given by $\max\{u_{i,j,1}, v_{i,j,1}\}$. The constraint is activated by $\gamma_{i,m}$, which is valid only if the job passes through the machine. Indeed, the term $\mathcal{B}$ represents a huge term, computed as the sum of all processing times, needed to activate or deactivate certain constraints.

Constraint (8.2b) ensures that only one machine per type is selected, i.e., one RMG, one PM, and one RS. Constraint (8.2c) ensures that containers follow the correct sequence through the RMG, PM, and then RS, meaning each container should be handled by the RMG first, then the PM, and finally the RS. Specifically, M_y refers to a set of machines for the y -th operation, and M_{y+1} refers to the set of machines for the subsequent operation.

Constraints (8.2d) and (8.2e) ensure that a container has at most one predecessor and one successor on the various machines. Given the possibility of multiple paths, constraint (8.2f) ensures that a set of machines has the correct number of precedence constraints, with $\#M_y$ denoting the cardinality of the set. Constraints (8.2g) and (8.2h) ensure the mutual exclusivity of jobs, while constraints (8.2i) and (8.2j) express the correct relationships among the decision variables x , γ , and δ .

Constraints (8.2k) and (8.2l) ensure that each machine handles at most one container. Constraint (8.2m) expresses the waiting time in relation to the completion time on the machine and the subsequent start time on the machines belonging to the next set. Finally, constraint (8.2n) is used to obtain the makespan value.

8.1.2.3 Input dataset

To assess the effectiveness of the proposed methodology, the discharge process of a single train at the PSA GP terminal is analyzed.

The actual discharge sequence adopted for this train is used as a benchmark, and the results obtained through the proposed methodology are compared with those generated by the operational approach currently followed by the terminal's rail operators.

The input data provided by the terminal for the selected train include information such as the assigned track, container characteristics, designated yard storage positions after discharge, and the operational speeds of the handling vehicles (RMG, PM, and RS). These real operational data are integrated with estimated values derived from a simulation model in order to compute the setup and processing times associated with each job on each machine. It should be noted that certain approximations are introduced in the estimation of these times, particularly for parameters derived from vehicle speeds. These approximations are necessitated by data limitations and by the requirement to estimate values that are not directly available from the operational database.

The train considered in this case study transports a limited number of containers, namely six, which correspond to six individual jobs in the scheduling formulation. The discharging opera-

Table 8.1: Processing times for each job on each machine (in seconds)

Machines	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
RMG	63	63	63	63	63	63
PM1	193	193	167	96	40	96
PM2	193	193	167	96	40	96
RS	60	65	70	60	60	65

Table 8.2: Set up times for each job on RMG (in seconds)

Jobs	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
Job 1	-	36	46	56	66	76
Job 2	36	-	36	46	56	66
Job 3	46	36	-	36	46	56
Job 4	56	46	36	-	36	46
Job 5	66	56	46	36	-	36
Job 6	76	66	56	46	36	-

tions are performed using one RMG, two PMs, and one RS.

The processing times, defined as described in the previous section, are computed on the basis of the operational speeds of the respective handling vehicles and the distances required to complete each operation. In the case of the PMs, these distances differ for each job, since each container is allocated to a distinct storage position within the yard. As a result, the travel distance from the train to the designated yard location varies from one container to another, directly influencing the corresponding processing time.

For the RMG, the processing time is determined by the trolley travel speed and the associated horizontal displacement, in addition to the hoisting speed and the vertical lifting distance required for each container. The setup times for the RMG when transitioning between jobs are presented in Table 8.2. These times primarily depend on the distance the gantry and trolley must travel to reach the next container, as well as the time required for the hoist to operate.

Similarly, the setup times associated with the RS, reported in Table 8.3, depend on the distances between the designated storage positions of the containers within the yard. After stacking one container, the RS is required to travel to the location of the subsequent container, and this relocation time is modeled as setup time.

In contrast, no setup time is assigned to the PMs in the considered scenario. This assumption is based on the premise that the RMG movement time is negligible compared to the travel time of the PMs. Therefore, when a PM completes the discharge of job i and proceeds to handle job j , the RMG is assumed to be available without additional delay.

Table 8.3: Set up times for each job on RS (in seconds)

Jobs	Job 1	Job 2	Job 3	Job 4	Job 5	Job 6
Job 1	-	0	115	174	263	174
Job 2	0	-	115	141	263	174
Job 3	115	115	-	63	151	63
Job 4	174	141	63	-	102	0
Job 5	263	263	151	102	-	102
Job 6	174	174	63	0	102	-

Nevertheless, the proposed modeling framework can be readily extended to include setup times for the PMs, should a more detailed representation of the operational dynamics be required.

8.1.2.4 Exact solution results

The tests were conducted in Python using the Gurobi mathematical optimization solver on a computer equipped with an Intel(R) Core(TM) Ultra i7-155H processor (3.80 GHz) and 16 GB of RAM.

Fig. 8.3 presents the solution implemented by the terminal to discharge the specific train. Jobs are represented in different colors: the thinnest hatched section indicates the setup time, the thickest section represents the processing time on each machine, and the cross-hatched section denotes the waiting time before a job is transferred to the next machine after processing. In contrast, Fig. 8.4 illustrates the job schedule for each machine as a result of the proposed scheduling methodology. In this specific case, the proposed methodology reduces the total train handling time by approximately 25%.

Furthermore, with the right input data, the model would automatically generate the optimal discharge schedule or sequence for the train, greatly assisting the rail operators in their work.

Given the NP-hard nature of the analyzed problem, a heuristic approach should be considered for high-dimensional cases to reduce computational time, which will be explored in future work. However, the proposed approach is well-suited for small to medium-sized problems, such as the scenario analyzed in this study, where solving the problem takes approximately 8 seconds. Additionally, for a huge number of containers, dynamic scheduling and rescheduling of containers can be implemented to partition the workload, focusing only on those with approaching due dates.

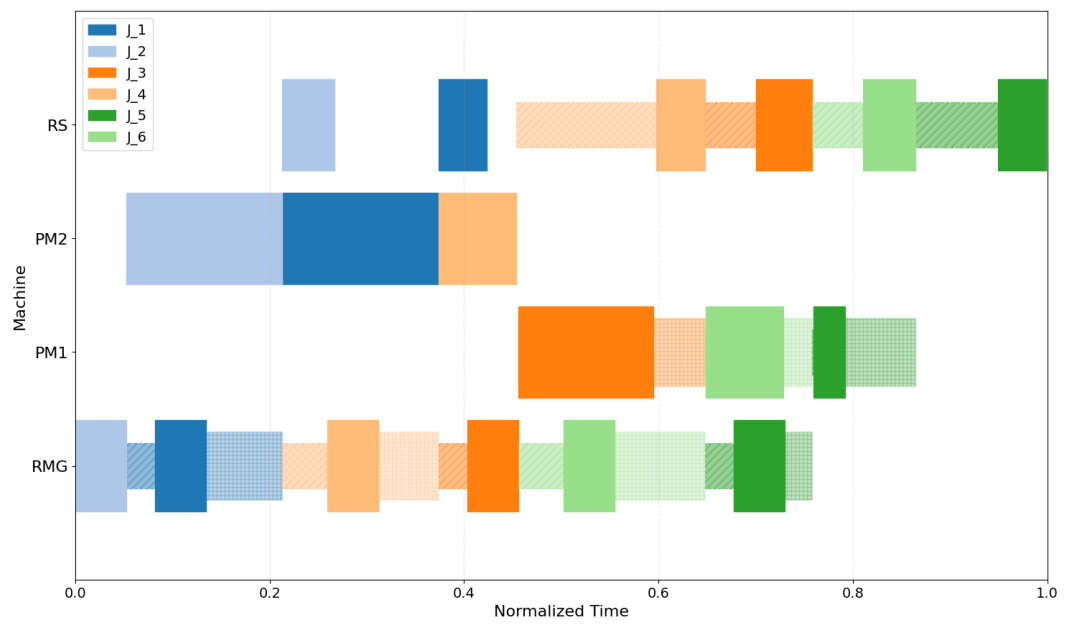


Figure 8.3: Solution performed by the terminal operators

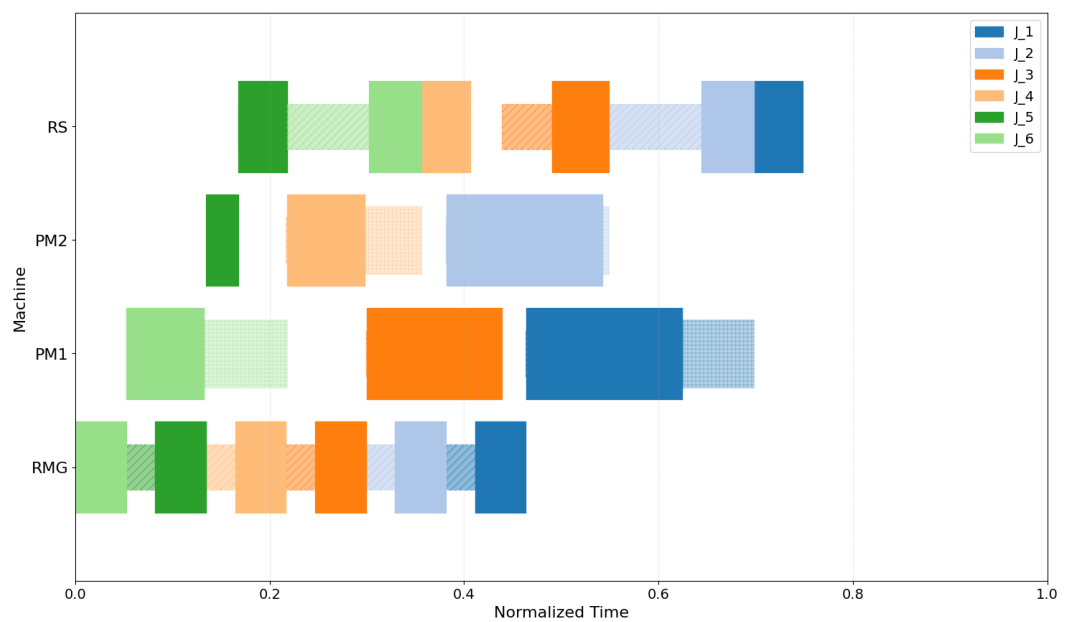


Figure 8.4: Optimal solution of the proposed methodology

8.1.3 Metaheuristic solution

The solution approach presented in Subsection 8.1.2 provides an exact solution within a very short computational time. However, this approach is effective only for relatively small problem instances, such as the one considered in the previous analysis. As the problem size increases, solving the model using this methodology becomes computationally challenging.

To address this limitation, an alternative solution approach is adopted for larger instances. In particular, a Tabu Search (TS) metaheuristic is implemented and tested to solve the problem for larger-scale scenarios. The formulation of the metaheuristic for the train discharge problem is presented in the following section.

8.1.3.1 Tabu search model

The train discharge planning problem is modeled as a three-stage blocking scheduling problem with sequence-dependent setup times. Each container corresponds to a job that must be processed in sequence by the main handling resources involved in the discharge operation: the RMG, the PMs, and the RS.

Jobs, stages, and resources:

Let $J = \{1, \dots, n\}$ denote the set of containers to be discharged from an inbound train. Each job $j \in J$ must pass through the following stages:

1. Stage 1 (RMG): the crane unloads the container from the rail wagon and places it onto a PM;
2. Stage 2 (PM transport): a PM transports the container from the rail area to its assigned storage location in the yard;
3. Stage 3 (RS stacking): the RS unloads the container from the PM and stacks it in the yard block.

The resource configuration consists of one RMG, a set of identical PMs (denoted M), and one RS. Each job is subject to the precedence constraints requiring Stage 1 to be completed before Stage 2, and Stage 2 before Stage 3.

Processing and setup times: Let $p_{j,m}$ denote the processing time of job j on resource m . Sequence-dependent setup times are considered: if job i is processed immediately before job j on resource m , a setup time $s_{i,j,m}$ is incurred before processing j (with $s_{*,j,m} = 0$ if j is the first job processed on resource m).

The definitions of processing, setup, and waiting times remain the same as those introduced in the FFSSP formulation. For the RMG, the setup time accounts for horizontal repositioning along

the train and spreader adjustments when required. For the RS, the setup time represents the travel time between consecutive stacking locations in the yard. In the present scenario, setup times are not considered for PMs; however, the framework can be extended to incorporate them if necessary.

Constraints:

The system operates under blocking conditions due to the absence of intermediate buffers between consecutive stages. In particular:

- After completing Stage 1, the RMG remains blocked holding job j until the transportation stage (Stage 2) begins on an assigned PM.
- After completing Stage 2, the PM remains blocked holding job j until the RS begins Stage 3 for that job.

Therefore, upstream resources may remain occupied while waiting for downstream resources to become available.

Decision variables:

A feasible solution is characterized by the following decision variables:

- A permutation $\pi^{(1)}$ representing the processing order of containers at the RMG.
- A permutation $\pi^{(3)}$ representing a priority list for the RS.

In this formulation, $\pi^{(3)}$ is not enforced as a strict sequence that could cause the RS to remain idle. Instead, it defines a ranking among jobs that are ready for Stage 3, and the RS selects the highest-ranked job among those available.

The assignment of PMs is not explicitly treated as a decision vector. Instead, the PM executing Stage 2 is determined dynamically by the schedule decoder using a greedy dispatching rule.

Objective function:

The objective is to minimize the makespan of the discharge process:

$$\min C_{\max} = \max_{j \in J} C_j^{(3)}, \quad (8.3)$$

where $C_j^{(3)}$ denotes the completion time of job j at Stage 3 (RS).

Minimizing $C_{\max}$ reduces the total time during which a train occupies a track in the rail park, thereby increasing track availability and improving overall rail terminal throughput and operational efficiency.

8.1.3.2 Solution method

Due to the combinatorial complexity induced by blocking constraints and sequence-dependent setup times, the problem is solved using a TS metaheuristic applied to the pair $(\pi^{(1)}, \pi^{(3)})$.

Schedule evaluation:

Each candidate solution $(\pi^{(1)}, \pi^{(3)})$ is evaluated by constructing a feasible schedule through a deterministic event-based simulation that enforces: (i) stage precedence constraints, (ii) machine capacity constraints, (iii) sequence-dependent setup times, and (iv) blocking constraints.

The simulation maintains the availability times of machines, the last job processed on each machine (for setup computation), and a priority queue of completion events. The simulation clock advances to the next event time t , all completions occurring at time t are processed, and the decoder subsequently starts all operations that can begin immediately at time t according to the following priority rules:

1. RS first (bottleneck-first): if the RS is idle and at least one job has completed Stage 2, the RS starts the ready job with the highest priority according to $\pi^{(3)}$.
2. Unblocking the RMG through Stage 2 activation: if the RMG is holding a completed Stage 1 job j , and a PM is available and not blocked, job j immediately starts Stage 2 on a PM selected according to a greedy earliest-completion rule.
3. Starting the next RMG job: if the RMG is free and unprocessed jobs remain, the next job according to $\pi^{(1)}$ begins processing on the RMG.

When a job starts Stage 2, the RMG is released at that start time. Similarly, when a job starts Stage 3, the corresponding PM is released.

Greedy PM selection:

When the RMG is blocked by a job j that has completed Stage 1, the decoder assigns j to an available PM by selecting the PM that minimizes the resulting Stage 2 completion time at the current time t .

Neighborhood structures:

The TS explores the solution space using two move types applied to $\pi^{(1)}$ and $\pi^{(3)}$:

- Insertion move: a job is removed from position i in a permutation and reinserted at position j .
- Filtered adjacent swap move: two consecutive jobs are swapped, but only if the swap reduces the local setup contribution.

More precisely, consider three consecutive jobs (k, i, j) processed on the same machine. Before the swap, the local contribution to the schedule length is

$$p_{i,m} + s_{k,i,m} + p_{j,m} + s_{i,j,m},$$

where $p_{i,m}$ and $p_{j,m}$ denote the processing times of jobs i and j on machine m , and $s_{a,b,m}$ represents the sequence-dependent setup time incurred when job b follows job a on machine m .

After swapping i and j , the local contribution becomes

$$p_{j,m} + s_{k,j,m} + p_{i,m} + s_{j,i,m}.$$

The swap $(k, i, j) \rightarrow (k, j, i)$ is accepted if

$$p_{j,m} + s_{k,j,m} + p_{i,m} + s_{j,i,m} < p_{i,m} + s_{k,i,m} + p_{j,m} + s_{i,j,m}.$$

Since $p_{i,m} + p_{j,m}$ appears on both sides of the inequality, the processing times cancel out. Therefore, the decision reduces to the setup-based condition

$$s_{k,j,m} + s_{j,i,m} < s_{k,i,m} + s_{i,j,m}.$$

This condition ensures that only locally improving swaps, in terms of sequence-dependent setup times, are considered in the neighborhood, thereby reducing the exploration of unpromising moves.

Tabu list:

To avoid cycling (i.e., repeatedly undoing the same changes), the TS maintains a *tabu list*, which represents a short-term memory of recently applied move attributes. After a move is performed, the corresponding attribute is forbidden for a fixed number of iterations (tabu tenure).

Tabu attributes are stored in a normalized manner: (i) for an insertion move, the moved job and the destination position are recorded (job j is temporarily forbidden from being inserted into position q), and (ii) for a swap move, the swapped index pair is recorded (positions (i, j) are temporarily forbidden from being swapped again).

This normalization prevents immediate reversals and short repetitive loops without excessively restricting the search. For example, if job 5 is inserted into position 2, reinserting job 5 into position 2 is forbidden for the next *tenure* iterations; similarly, if positions $(3, 4)$ are swapped, swapping $(3, 4)$ again is temporarily forbidden.

A tabu move may still be accepted under an *aspiration* rule: if the move produces a makespan smaller than the best solution found so far, the move is accepted even if it is tabu.

For efficiency, decoded schedules are cached using the pair $(\pi^{(1)}, \pi^{(3)})$. Furthermore, when the neighborhood becomes very large, only a random subset of at most 250 candidate moves is evaluated at each iteration in order to limit computational effort.

Diversification strategy:

If no improvement is observed for a predefined number of iterations, a diversification mechanism is applied by performing a small number of random swaps in both $\pi^{(1)}$ and $\pi^{(3)}$. The first solution producing a feasible decoded schedule is accepted; otherwise, the search returns to the best solution found so far.

8.1.3.3 Metaheuristic solution results

The proposed methodology is applied to a case study involving the discharge of a train at PSA GP, consisting of 24 containers. The results obtained from the model are compared with the operational sequence used by the terminal. For benchmarking purposes, the actual discharge sequence implemented in practice is considered; however, the real operational timestamps are not used. Instead, both the operational sequence and the proposed solution are evaluated under a common set of input parameters to ensure a consistent and fair comparison.

The model inputs include the processing times of each job on each machine and the sequence-dependent setup times for the RMG and the RS. These times are not taken directly from historical operational records. Rather, they are computed based on estimated equipment speeds and travel distances. The distance and velocity parameters used in these calculations are approximations of real operational values, obtained from a simulation model of the terminal.

In Fig. 8.5, the scheduling sequence followed by the terminal operators to discharge the specific train is illustrated. In this example, a set of five PMs is deployed. Jobs are represented using different visual patterns: the thin hatched segments indicate the setup time, the thick segments represent the processing time on each machine, and the cross-hatched segments denote the waiting time before a job is transferred to the subsequent machine after processing.

Figs. 8.6 and 8.7 present the schedules obtained using the proposed metaheuristic under different PM configurations.

Performance with the Same Number of PMs

When the metaheuristic is applied using the same number of PMs as in the real terminal configuration (5 PMs), the makespan is reduced by 10.259% compared to the operational sequence.

This improvement is primarily due to a better synchronization between the RMG, PMs, and the RS, which reduces unnecessary waiting times and improves the utilization of the bottleneck resource. The results indicate that the operational sequence currently used in practice does not fully exploit the coordination potential among resources.

Impact of Reducing the Number of PMs

To further analyze the system behavior, additional experiments were conducted by reducing the number of PMs.

First, the model was executed with 3 PMs. The obtained makespan remains the same with the metaheuristic operational solution that uses 5 PMs. This result highlights that increasing the number of PMs beyond a certain level does not significantly impact the global performance. Instead, the RS processing stage constitutes the main bottleneck affecting the makespan.

Finally, the metaheuristic was tested with only 2 PMs. Even under this reduced resource configuration, the makespan is improved by 8.037% compared to the initial operational solution (with 5 PMs).

This finding is particularly relevant from a managerial perspective: not only does the metaheuristic improve the objective function, but it also enables a reduction in resource utilization while maintaining superior performance.

Computational Performance

The computational time required to obtain each metaheuristic solution is approximately 2 minutes on a standard workstation. Given the size of the instance (24 containers and sequence-dependent setup times), this runtime confirms the efficiency of the proposed approach.

The computational effort remains moderate and suitable for practical decision-support applications in terminal operations, where rapid generation of high-quality schedules is essential.

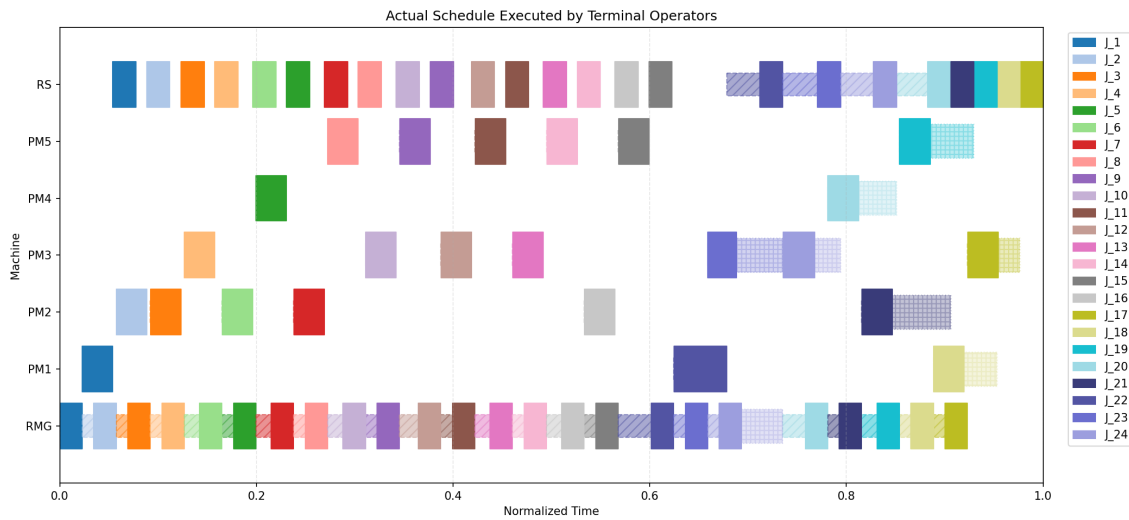


Figure 8.5: Scheduling of containers by terminal operators

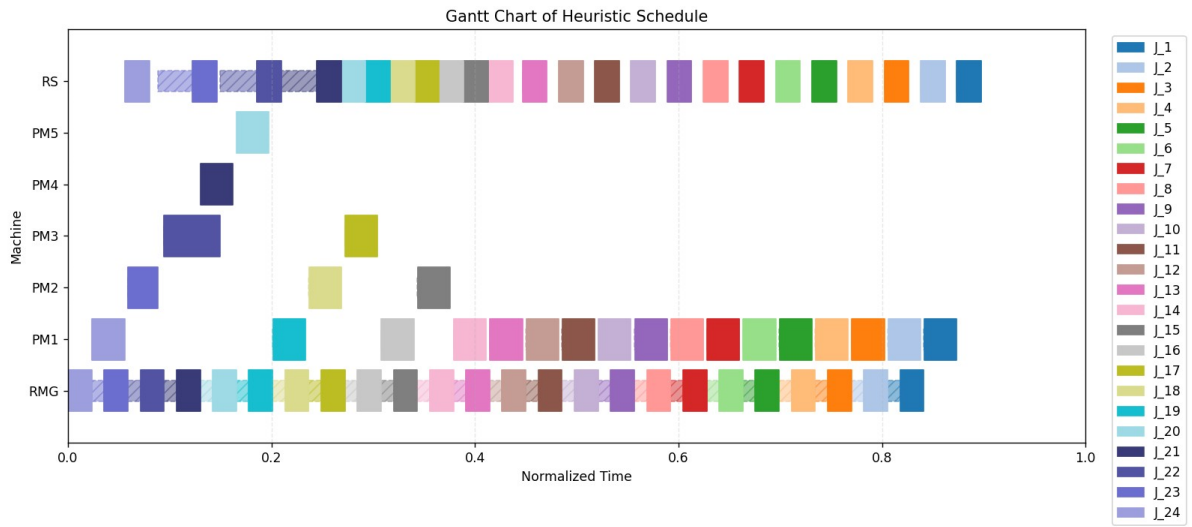


Figure 8.6: Metaheuristic scheduling of containers (5 PMs)

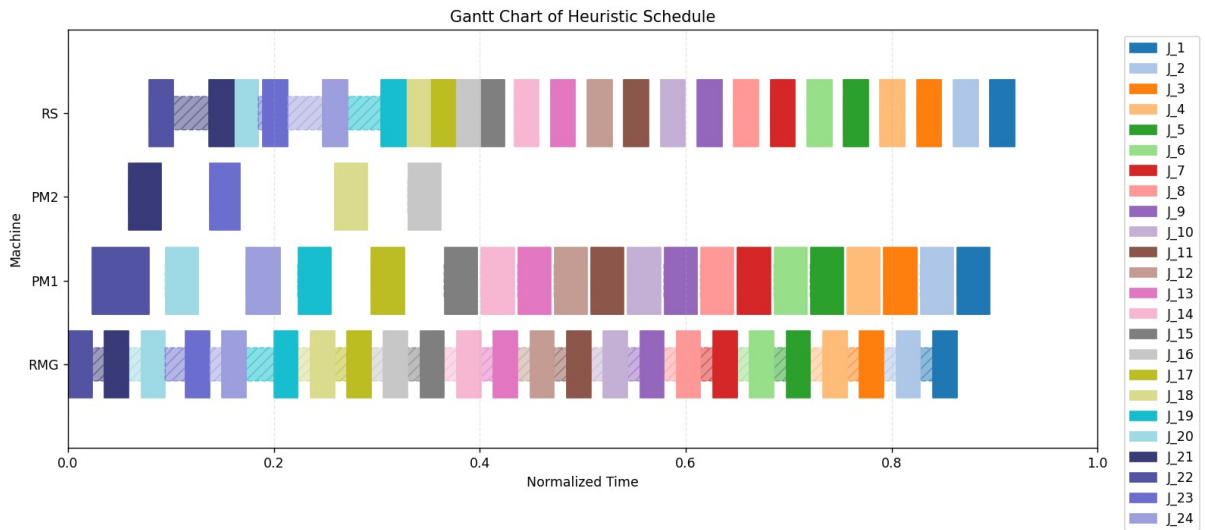


Figure 8.7: Metaheuristic scheduling of containers (2 PMs)

8.2 Simulation of train discharge and loading

The train discharge and loading processes are also modeled within FlexTerm. A detailed simulation model of the rail park has been developed, including the yard area, while other interfacing

components are abstracted in accordance with the FlexTerm modeling paradigm described in Section 5.3.6.1.

This section first presents the physical components that constitute the FlexTerm model and then describes the main logics and algorithms governing the behavior of the system and its objects. Fig. 8.8 illustrates the complete simulation model of the terminal developed in FlexTerm, including all its components.

Given the scope of the study in this thesis, the primary focus is on the rail yard. Therefore, the physical components and the logics described in the following sections mainly refer to this part of the model. Nevertheless, it is necessary to include certain yard components and related operational logics, as they play a fundamental role in the handling of containers during train discharge and loading operations.



Figure 8.8: Simulation model of the PSA GP terminal in FlexTerm

8.2.1 Physical components of the model

The main physical components of the detailed rail yard simulation model are described in this section. These components are also illustrated in Fig. 8.9.

- RMG: Represented by the built-in FlexTerm object *Gantry Crane*, an overhead crane that operates over a rail block to lift and move containers;
- RS: Represented by the built-in FlexTerm object *Top Loader*, a mobile crane used to handle

containers within the yard, typically for loading and unloading trucks within a given yard block;

- PM: Represented by the built-in FlexTerm object *Truck*, which transports containers (flow items) within the model. It becomes available when a Truck Gang is included in the system;
- Yard Areas: Represented by the built-in FlexTerm object *Yard Block*, a storage area for containers composed of cells arranged in rows and columns, where containers are stacked vertically;
- Rail Tracks: Represented by the built-in FlexTerm object *Rail Block*, which corresponds to a single train track area where trains arrive and depart;
- Train: Objects that appear in the model when defined in the rail planner, where their schedule and composition are specified;
- Network Lanes: Represented in FlexTerm by the *Multi Lane Object (MLO)*, part of the Travel Manager (TM). This object models road segments used by travelers controlled by the TM;
- Parking Spots: Designated locations where a traveler can stop. When a traveler is directed to a parking spot, a high-level path is generated based on the Multi Lane network formed by the MLOs;
- Advanced Rail Dispatcher: Represented in the model by the Dispatcher object, which is used to assign specific handling jobs in this case, containers to be processed to the RMG.

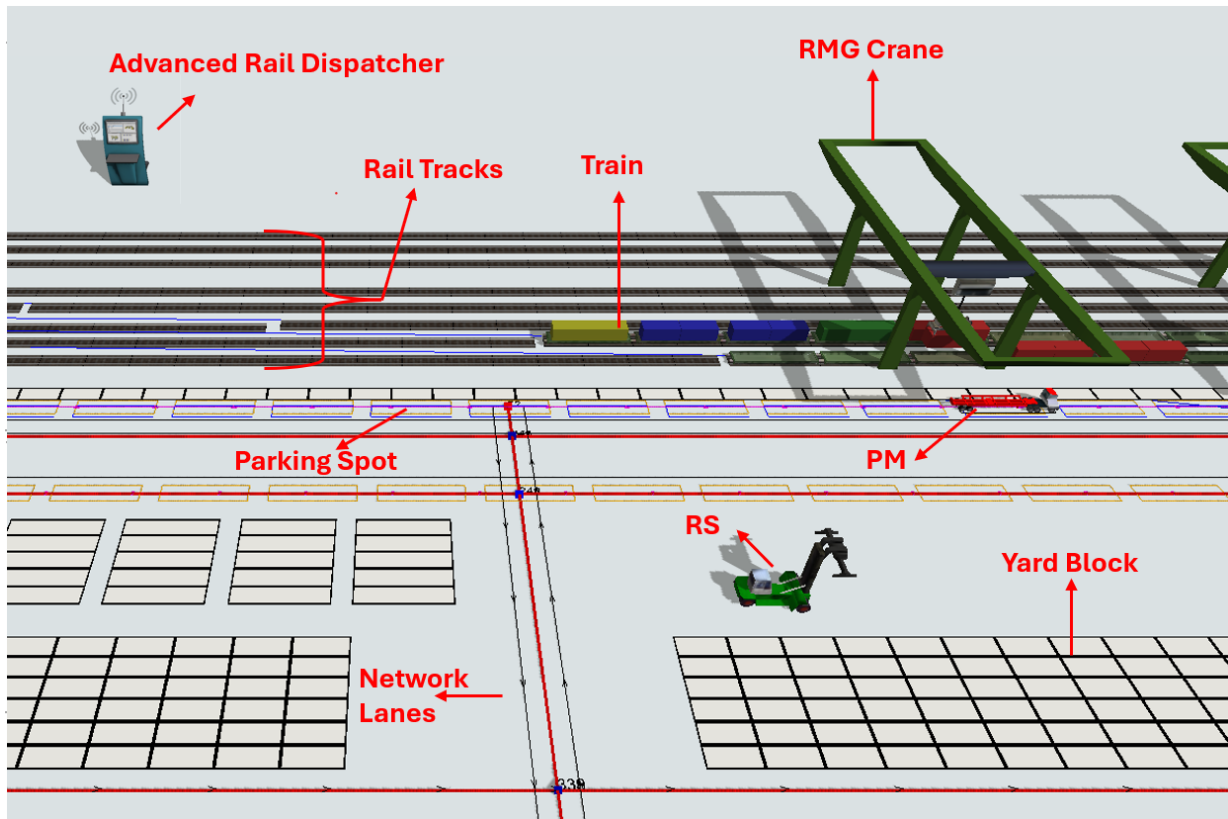


Figure 8.9: Main physical components for the modeling of rail operations in FlexTerm

It should be noted that most of these objects are equipped with built-in logic that enables them to operate realistically and ensures proper interaction within the system, without the need for additional implementation. This includes, for example, the movement logic of RMGs, RSs, and PMs, as well as collision avoidance between trucks, which is handled within the TM.

The following sections describe the additional logics implemented in the model, based on the current operational practices of the PSA GP terminal.

8.2.2 Logic behind the rail planner

As already mentioned in Section 5.3.6.2, the characteristics of incoming trains are defined through the logic of the Rail Planner. These include the number of wagons or cars, their types referred to as car profiles, which specify the type of containers they carry, the corresponding destination within the yard, and the train arrival schedule at the terminal.

Additional logic has been implemented to register handling jobs when a train enters the rail track.

Although the train arrives with a predefined composition from the Rail Planner, it is necessary to structure this information to enable efficient job dispatching. Specifically, each job is recorded in dedicated tables along with its relevant attributes, such as the container ID, the assigned rail track, the corresponding rail car, and the position within the car, referred to in FlexTerm as the bay, since a single rail car may carry multiple containers.

This registration process distinguishes between loading and discharge operations, based on the predefined load and discharge lists.

8.2.3 Advanced rail dispatcher logic

The core logic governing job dispatching and work strategy is implemented within the *Message Trigger* of the rail dispatcher. This logic handles different operational cases depending on the system state.

The first activation of this trigger occurs when a train arrives at a specific track. In this phase, the RMG assigned to that track is identified as the resource responsible for handling the train. Subsequently, additional messages are sent to the rail dispatcher to evaluate the status of the RMG, in particular whether it is idle or currently processing a job. When the RMG becomes available, its status is updated accordingly so that it can be considered for new assignments.

The availability of the RMG is communicated through the work strategy trigger of the Rail Planner, which continuously checks whether the crane is busy. If the RMG is idle, a message is sent to the rail dispatcher indicating that it is ready to process a new job.

Upon receiving this message, the rail dispatcher selects the next job to be executed. The selection is performed using a priority rule that reflects current terminal practices. Specifically, jobs are ordered based on the destination yard location, the container type (full or empty), and the container size (20-foot or 40-foot). This step represents a key area for improvement, as the current rule-based logic could be replaced by a more advanced approach derived from optimal or near-optimal solutions, as discussed earlier in this chapter.

Once a job is selected, the dispatcher verifies again the status of the RMG. If the crane is already assigned to a task, relevant information such as job completion time and crane assignment is updated. Otherwise, a new task sequence is generated for the PM, which is then dispatched to execute the job.

Finally, when all jobs associated with a train have been completed, a message is sent by the RMG to indicate that operations are finished. At this point, the train is allowed to depart, and all related jobs are removed from the corresponding tables to prevent reassignment.

8.2.4 Travel manager and parking spot selection strategy

TM and its functionalities play a key role in facilitating the movement of travelers within the terminal. Its integration into the model involves several steps; however, the focus in this section is on the selection of parking spots for PMs. This ensures that a PM stops at the appropriate location, either to pick up a container from a train or to deliver a container for loading onto a train.

The parking spot selection is based on the position of the container on the train. For each container being transported to or from the train, the corresponding car identifier (*carid*) is first retrieved. From the *carid*, the bay number is determined, representing the position of the container within the rail car. This information is then used to compute the spatial coordinates (e.g., x and y) of the container location.

Subsequently, among all available parking spots along the lane adjacent to the rail yard, the one closest to the computed bay position is selected. More specifically, a distance criterion is applied: a parking spot is considered suitable if the distance between its position and the bay location is less than or equal to a predefined threshold. The parking spot satisfying this condition is then assigned to the PM.

The parking spot selection procedure is summarized in Algorithm 4.

Algorithm 4 Parking Spot Selection for PMs

Require: Container information and train assignment

Ensure: Parking spot assigned to the PM

- 1: Retrieve the container destination
 - 2: **if** the container is to be discharged **then**
 - 3: Retrieve the corresponding car identifier from the incoming train data
 - 4: **else if** the container is to be loaded **then**
 - 5: Retrieve the corresponding car identifier from the outgoing train data
 - 6: **end if**
 - 7: Retrieve the train identifier
 - 8: Determine the track assigned to the train
 - 9: Determine the bay corresponding to the selected car
 - 10: Compute the container position on the rail track
 - 11: **for** each parking spot along the lane adjacent to the rail yard **do**
 - 12: Compute the distance between the parking spot and the container position
 - 13: **if** the distance is within the predefined threshold **then**
 - 14: Assign the parking spot to the PM
 - 15: **return** selected parking spot
 - 16: **end if**
 - 17: **end for**
-

8.2.5 Limitations and future work

It is important to note that a considerable effort was devoted to the development of this part of the research. The proposed model and the implemented logic are complete and function as intended, accurately representing the underlying problem structure.

However, the full validation of the model and the execution of a comprehensive set of comparative experiments, intended to contrast the scheduling strategy currently used by the terminal with the schedule proposed in this work, require additional effort. In particular, the simulation environment adopted in this study incorporates a more detailed and realistic representation of the terminal, including accurate travel distances, equipment velocities, and operational constraints that were necessarily simplified in the optimization formulation.

Conducting a thorough experimental campaign under these more realistic conditions would allow for a more precise assessment of the impact of the proposed methodology, but it is time-intensive and therefore beyond the scope of the present work. For this reason, these validation activities are identified as an important direction for future research.

Related publications. Parts of the work presented in this chapter have been published in [123]. Furthermore, the work related to the optimization of train discharge through a Tabu Search formulation is presented in [124], which has been accepted for presentation at an international conference.

Chapter 9

Rail Yard Capacity Analysis

Continuing with the rail subsystem, the final part of the research objective focuses on the evaluation of the capacity of the terminal's rail facility, as illustrated in Fig. 9.1.

The analysis begins with a detailed study of the physical rail facility, which serves as the basis for system modeling. Two complementary approaches are adopted to achieve this objective. The first approach consists of developing a simplified mathematical model of the system. Due to the complexity of real rail operations, this analytical representation requires the introduction of simplifying assumptions and therefore provides a theoretical evaluation of the overall capacity of the facility in terms of the number of trains that can be handled.

The second approach is based on the use of discrete-event simulation software, FlexTerm, to model the physical rail yard and the processes associated with train entry and exit. This detailed simulation model enables a more realistic assessment of the terminal's rail capacity by capturing operational dynamics and allowing the evaluation of the number of trains that can be handled within a given time period.

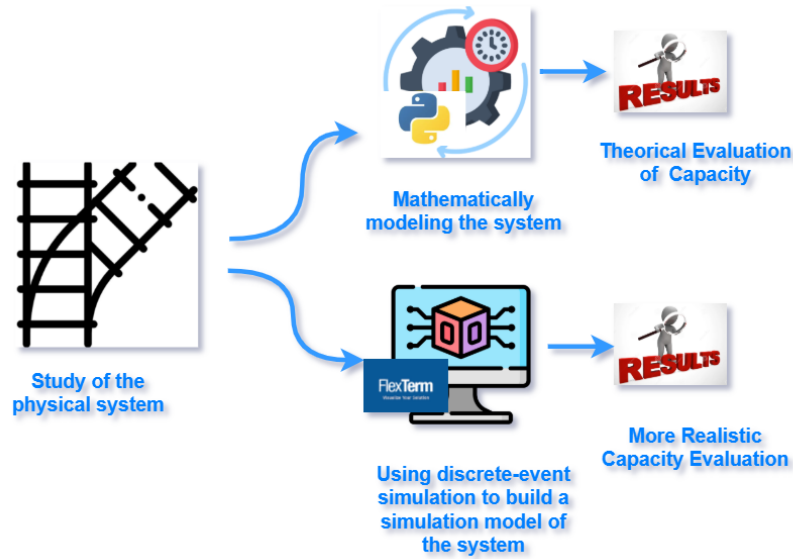


Figure 9.1: Conceptual framework of the research for rail capacity evaluation

9.1 Evaluation of theoretical rail yard capacity

As previously stated, the main objective of this study is to assess the capacity of the PSA Genova Pra' (PSA GP) rail facility, expressed as the maximum number of trains that can be handled within a given planning horizon.

To this end, queuing theory is initially adopted as an analytical framework. Each track within the rail parks is modelled as an individual queue, where arriving trains wait to be processed according to the available service capacity. In this context, the service rate reflects the operational productivity of the handling resources assigned to each track.

9.1.1 General description of the rail park

The rail yard within the terminal extends approximately 950 meters and consists of 8 internal tracks, divided into two main sections: the Ponente Park (West Park) and the Levante Park (East Park). The tracks in the Ponente Park have lengths ranging from 420 to 536 meters, whereas those in the Levante Park range from 420 to 560 meters.

Upon arrival at the terminal, trains initially stop on one of the five external tracks, while a sixth external track is reserved exclusively for locomotive transit (highlighted in red in Fig. 9.2). At this stage, the main line locomotive is decoupled and replaced by a shunting locomotive. The

latter then pushes the train toward one of the two internal parks through dedicated connecting tracks, hereafter referred to as intermediate tracks.

In order to access the east tracks, a train must necessarily pass through specific west tracks. Consequently, at least one designated west track must be available. This availability is not interchangeable, as the required west track depends on the final destination track assigned to the train.

Referring to Fig. 9.2, which illustrates the layout of the railway park, a train can move to track E_8 only if track W_8 is free at that moment. Similarly, access to E_7 requires either W_8 or W_7 to be free. This pattern continues down to E_1 , where the train can proceed as long as at least one of the west tracks, from W_1 to W_8 , is available.

There are four available Rail-Mounted Gantry (RMG) cranes serving the internal tracks, two at each park. Currently, however, terminal operations proceed with only one RMG per park during each shift. In this context, a shift refers to a designated 6-hour work period during which a specific crew is on duty.

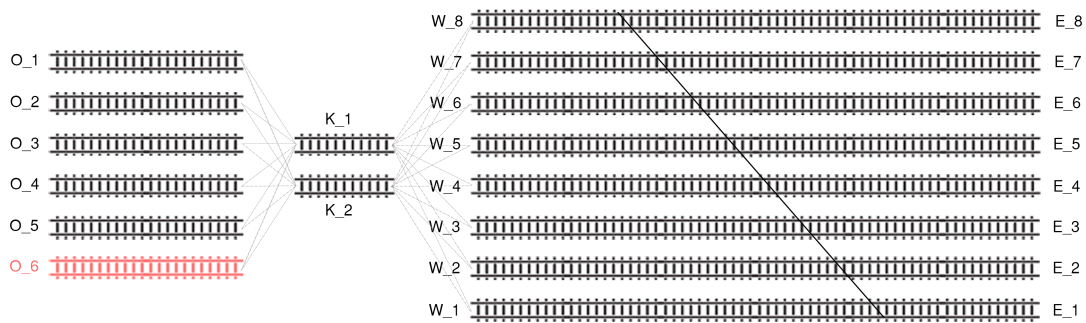


Figure 9.2: Railway park overview

In the following subsections, the mathematical formulation of the problem is presented. First, the optimization model is formulated and solved without considering locomotive constraints, assuming that a locomotive is always available whenever required. Subsequently, the same problem is reformulated and solved by explicitly incorporating locomotive-related constraints, including the limited number of available locomotives and the requirement for locomotives to support train entrance and exit movements.

9.1.2 Process modeling without locomotive constraints

Building on the physical and operational characteristics outlined in the previous subsection, an optimization model is formulated that incorporates the modeling aspects necessary to represent the process under consideration. The parameters of the problem are defined as follows:

- N_O : the number of external tracks;
- N_W : the number of west park tracks;
- N_E : the number of east park tracks;
- N_K : the number of intermediate tracks;
- N_G : the number of RMG cranes in each park;
- P_{RMG} : the productivity of the RMG cranes;
- T : the interval of time taken into consideration.

In the problem formulation, when defining constraints that apply to all internal tracks (both west and east), the total number of tracks is denoted as N_I , where $N_I = N_W + N_E$.

The decision variables are the following binary quantities:

- $q_i^O(t)$: the queue length, expressed as the number of trains on external track i at time t ;
- $q_j^W(t)$: the queue length, expressed as the number of trains on internal west park track j at time t ;
- $q_j^E(t)$: the queue length, expressed as the number of trains on internal east park track j at time t ;
- $a_{i,j}(t)$: the arrival rate of trains at external track i for internal track j at time t ;
- $y_{i,j,k}(t)$: equals 1 if a train moves from external track i to internal track j via intermediate track k at time t , and 0 otherwise;
- $F_j(t)$: equals 1 if the train on internal track j has completed its operation at time t , and 0 otherwise;
- $y'_{i,j,k}(t)$: equals 1 if a train, after finishing its operation, moves from internal track j to external track i via intermediate track k at time t , and 0 otherwise;

- $\delta_{i,j}(t)$: equals 1 if, at time $t - 1$, a train moving from external track i to internal track j could not enter due to the unavailability of a free intermediate track or a free west track. It serves to track trains that are waiting to enter and are blocked at the previous time step;
- $z_j(t)$: a variable representing the minimum between 1 and the number of free west tracks that allow a train for the east track j to enter or exit;
- $g_j^W(t)$: equals 1 if an RMG crane is assigned and operating on west park track j at time t , and 0 otherwise;
- $g_j^E(t)$: equals 1 if an RMG crane is assigned and operating on east park track j at time t , and 0 otherwise;
- $r_j(t)$: a binary variable that becomes 1 as soon as the train on internal track j finishes processing. It remains 1 until the train exits the system. This variable helps track completed trains that are ready to depart but are temporarily delayed due to the unavailability of intermediate, external, or west tracks.

9.1.2.1 Constraints without locomotive considerations

The objective function to be maximized is the number of trains that the terminal can complete or handle within a specified time horizon T . The statement of the resulting optimization process is the following.

Problem 7

$$\max \sum_{i=1}^{N_O} \sum_{j=1}^{N_I} \sum_{k=1}^{N_K} \sum_{t=1}^T y'_{i,j,k}(t) \quad (9.1)$$

subject to

$$\sum_{j=1}^{N_I} a_{i,j}(t) \leq 1 \quad \forall t = 1, \dots, T \quad \forall i = 1, \dots, N_O \quad (9.2)$$

$$\sum_{i=1}^{N_O} a_{i,j}(t) \leq 1 \quad \forall t = 1, \dots, T \quad \forall j = 1, \dots, N_I \quad (9.3)$$

$$\sum_{i=1}^{N_O} \sum_{j=1}^{N_I} y_{i,j,k}(t) + \sum_{i=1}^{N_O} \sum_{j=1}^{N_I} y'_{i,j,k}(t) \leq 1 \quad \forall k = 1, \dots, N_K \quad \forall t = 1, \dots, T \quad (9.4)$$

$$\begin{aligned} y_{i,j,k}(t+1) &\leq q_i^O(t) \quad \forall i = 1, \dots, N_O \\ \forall j = 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad \forall t = 1, \dots, T-1 \end{aligned} \quad (9.5)$$

$$y_{i,j,k}(1) = 0 \quad \forall i = 1, \dots, N_O \quad \forall j = 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad (9.6)$$

$$\begin{aligned} y'_{i,j,k}(t) &\leq q_j^W(t-1) \quad \forall i = 1, \dots, N_O \\ \forall j = 1, \dots, N_W \quad \forall k = 1, \dots, N_K \quad \forall t = 2, \dots, T \end{aligned} \quad (9.7)$$

$$\begin{aligned} y'_{i,j,k}(t) &\leq q_{j-N_W}^E(t-1) \quad \forall i = 1, \dots, N_O \\ \forall j = N_W + 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad \forall t = 2, \dots, T \end{aligned} \quad (9.8)$$

$$\delta_{i,j}(1) = 0 \quad \forall i = 1, \dots, N_O \quad \forall j = 1, \dots, N_I \quad (9.9)$$

$$\begin{aligned} \delta_{i,j}(t+1) &= \delta_{i,j}(t) + a_{i,j}(t) - \sum_{k=1}^{N_K} y_{i,j,k}(t+1) \\ \forall i = 1, \dots, N_O \quad \forall j = 1, \dots, N_I \quad \forall t = 1, \dots, T-1 \end{aligned} \quad (9.10)$$

$$\begin{aligned} \sum_{k=1}^{N_K} y_{i,j,k}(t+1) &\geq \delta_{i,j}(t) \quad \forall i = 1, \dots, N_O \\ \forall j = 1, \dots, N_I \quad \forall t = 1, \dots, T-1 \end{aligned} \quad (9.11)$$

$$\begin{aligned} F_j(t) &= \left\lfloor \sum_{\left(t - \frac{1}{P_{RMG}}\right) \leq t' \leq t} P_{RMG} * g_j^W(t') \right\rfloor \\ \forall j = 1, \dots, N_W \quad \forall t = 1, \dots, T \end{aligned} \quad (9.12)$$

$$F_j(t) = \left[\sum_{\left(t - \frac{1}{P_{RMG}}\right) \leq t' \leq t} P_{RMG} * g_{j-N_W}^E(t') \right] \quad (9.13)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$q_i^O(t+1) = q_i^O(t) + \sum_{j=1}^{N_I} a_{i,j}(t+1) - \sum_{j=1}^{N_I} \sum_{k=1}^{N_K} y_{i,j,k}(t+1) + \quad (9.14)$$

$$\sum_{j=1}^{N_I} \sum_{k=1}^{N_K} y'_{i,j,k}(t+1) - \sum_{j=1}^{N_I} \sum_{k=1}^{N_K} y'_{i,j,k}(t)$$

$$\forall i = 1, \dots, N_O \quad \forall t = 1, \dots, T-1$$

$$q_i^O(1) = \sum_{j=1}^{N_I} a_{i,j}(1) \quad \forall i = 1, \dots, N_O \quad (9.15)$$

$$q_j^W(t+1) = q_j^W(t) + \sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y_{i,j,k}(t+1) - \sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y'_{i,j,k}(t+1) \quad (9.16)$$

$$\forall j = 1, \dots, N_W \quad \forall t = 1, \dots, T-1$$

$$q_j^W(1) = 0 \quad \forall j = 1, \dots, N_W \quad (9.17)$$

$$q_j^E(t+1) = q_j^E(t) + \sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y_{i,j+N_W,k}(t+1) - \sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y'_{i,j+N_W,k}(t+1) \quad (9.18)$$

$$\forall j = 1, \dots, N_E \quad \forall t = 1, \dots, T-1$$

$$q_j^E(1) = 0 \quad \forall j = 1, \dots, N_E \quad (9.19)$$

$$\sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y_{i,j,k}(t) \leq \min(1, \left(\sum_{m=j-N_W}^{N_W} (1 - q_m^W(t)) \right)) \quad (9.20)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$\sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y'_{i,j,k}(t) \leq \min(1, (\sum_{m=j-N_W}^{N_W} (1 - q_m^W(t)))) \quad (9.21)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$r_j(t) = r_j(t-1) - \sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y'_{i,j,k}(t) + F_j(t) \quad (9.22)$$

$$\forall j = 1, \dots, N_I, \quad \forall t = 2, \dots, T$$

$$y'_{i,j,k}(t+1) \leq r_j(t) \quad \forall i = 1, \dots, N_O \quad (9.23)$$

$$\forall j = 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad \forall t = 1, \dots, T-1$$

$$g_j^W(t) \leq 1 - r_j(t-1) \quad (9.24)$$

$$\forall j = 1, \dots, N_W \quad \forall t = 2, \dots, T$$

$$g_{j-N_W}^E(t) \leq 1 - r_j(t-1) \quad (9.25)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 2, \dots, T$$

$$g_j^W(t) \leq q_j^W(t) \quad \forall j = 1, \dots, N_W \quad \forall t = 1, \dots, T \quad (9.26)$$

$$g_j^E(t) \leq q_j^E(t) \quad \forall j = 1, \dots, N_E \quad \forall t = 1, \dots, T \quad (9.27)$$

$$\sum_{j=1}^{N_W} g_j^W(t) \leq N_G \quad \forall t = 1, \dots, T \quad (9.28)$$

$$\sum_{j=1}^{N_E} g_j^E(t) \leq N_G \quad \forall t = 1, \dots, T \quad (9.29)$$

$$g_j^W(t) \leq 1 \quad \forall j = 1, \dots, N_W, \quad \forall t = 1, \dots, T \quad (9.30)$$

$$g_j^E(t) \leq 1 \quad \forall j = 1, \dots, N_E, \forall t = 1, \dots, T \quad (9.31)$$

$$g_j^W(t) \geq g_j^W(t-1) - F_j(t-1) \quad (9.32)$$

$$\forall j = 1, \dots, N_W \quad \forall t = 2, \dots, T$$

$$g_j^E(t) \geq g_j^E(t-1) - F_{j+N_W}(t-1) \quad (9.33)$$

$$\forall j = 1, \dots, N_E \quad \forall t = 2, \dots, T$$

Linearizing the floor and minimum function:

$$F_j(t) \leq \sum_{\left(t - \frac{1}{P_{RMG}}\right) \leq t' \leq t} P_{RMG} * g_j^W(t') \quad (9.34)$$

$$\forall j = 1, \dots, N_W \quad \forall t = 1, \dots, T$$

$$F_j(t) \geq \left(\sum_{\left(t - \frac{1}{P_{RMG}}\right) \leq t' \leq t} P_{RMG} * g_j^W(t') \right) - 1 + \epsilon \quad (9.35)$$

$$\forall j = 1, \dots, N_W \quad \forall t = 1, \dots, T$$

$$F_j(t) \leq \sum_{\left(t - \frac{1}{P_{RMG}}\right) \leq t' \leq t} P_{RMG} * g_{j-N_W}^E(t') \quad (9.36)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$F_j(t) \geq \left(\sum_{\left(t - \frac{1}{P_{RMG}}\right) \leq t' \leq t} P_{RMG} * g_{j-N_W}^E(t') \right) - 1 + \epsilon \quad (9.37)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$z_{j-W}(t) \leq \sum_{m=j-N_W}^{N_W} (1 - q_m^W(t)) \quad (9.38)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$z_{j-W}(t) \geq \frac{(\sum_{m=j-N_W}^{N_W} (1 - q_m^W(t)))}{M} \quad (9.39)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$\sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y_{i,j,k}(t) \leq z_{j-N_W}(t) \quad (9.40)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

$$\sum_{i=1}^{N_O} \sum_{k=1}^{N_K} y'_{i,j,k}(t) \leq z_{j-N_W}(t) \quad (9.41)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall t = 1, \dots, T$$

Constraint (9.2) guarantees that at any given time t , only one train can arrive at an external track i . Similarly, Constraint (9.3) enforces the same condition for the internal tracks j . Constraint (9.4) ensures that two trains cannot simultaneously enter and exit an intermediate track k at the same time t . Specifically, for any given track k , no train can enter and exit via the same track at the same time. Constraints (9.5) and (9.6) ensure that a train can enter the internal tracks only if it was previously waiting in the external queue. For time instant $t = 1$, this value is set to zero, as in this formulation, a train can enter at most one time step after its arrival.

Constraints (9.7), (9.8) enforce that a train can exit only if it has previously been on the internal tracks being operated on. Constraints (9.9) and (9.10) define the variable δ , which represents trains that have arrived but are still waiting to enter the terminal. These are trains that do not enter the internal tracks immediately upon arrival, due to the unavailability of intermediate or internal tracks, and are therefore stored in δ .

Constraint (9.11) ensures that trains waiting in the external tracks are given priority to enter at time t over newly arriving trains at the same time.

Constraints (9.12) and (9.13) model the completion status of trains on each internal track. In this formulation, a train is considered completed once a sufficient amount of processing time has elapsed. This processing time depends on P_{RMG} , which represents the percentage of the

train handled per discrete time unit, i.e., the productivity of an RMG crane. The binary variable $F_j(t)$ takes value 1 when the train is completed and returns to 0 afterwards. Since the presence of the floor operator $\lfloor \cdot \rfloor$, which returns the greatest integer less than or equal to its argument, makes the model nonlinear, Constraints (9.12) and (9.13) are reformulated using an equivalent linear representation, as reported in Constraints (9.34)–(9.37). The parameter ϵ denotes a small positive constant introduced to properly activate or deactivate the constraint when required.

Constraints (9.14)–(9.19) describe the dynamics of each queue: external, west, and east. Constraints (9.20) and (9.21) represent the previously described physical constraints governing train movements toward or from the east tracks. As discussed, these movements are feasible only if at least one of the designated west tracks is available.

Since the presence of the $\min(\cdot)$ operator introduces nonlinearity into the model, Constraints (9.20) and (9.21) are reformulated through an equivalent linear representation. The corresponding linearization is provided by Constraints (9.38)–(9.41).

Constraint (9.22) is introduced to track trains that have been completed for more than one time step. Since $F_j(t)$ becomes 1 only at the moment a train is completed and then immediately returns to 0, this new variable is used to remain equal to 1 after completion, indicating that the train is finished and ready to depart, and returns to 0 only when the train actually leaves the internal track. Then, constraint (9.23) ensures that a train can leave the internal track at any time step after it has completed its operations.

Constraints (9.24) and (9.25) prevent a crane from being reassigned to a train that has already been finished but is still waiting to depart. Constraints (9.26) and (9.27) ensure that a crane is assigned only to tracks where a train is present and requires processing. Constraints (9.28) and (9.29) ensure that no more than the available number of RMG cranes are assigned to the tracks in each park. Constraints (9.30) and (9.31) ensure that each crane is assigned to only one track at a time. According to the current logic, an RMG can work on only one train at a time. Meanwhile, constraints (9.32) and (9.33) ensure that the crane remains active on a specific track for the necessary duration.

9.1.2.2 Results without locomotive constraints

A numerical campaign was conducted to provide the basis for a broader analysis aimed at evaluating the capacity of the rail yard. The experiments were performed in Python using the Gurobi mathematical optimization solver.

Specifically, the values of several key parameters were varied in order to observe their impact on the capacity of the rail yard and to identify which parameters have the greatest influence on system performance. The first set of experiments considers variations in the number of intermediate tracks, combined with scenarios in which either one or two RMG cranes are operating in both the East and West parks. In this analysis, the configuration of the external tracks ($N_O = 5$)

Table 9.1: Number of trains processed for each (N_K, N_G) combination over time horizons $t = 1$ to 24

(T)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$N_K=1, N_G=1$	0	0	0	0	0	0	1	2	2	2	2	3	4	4	4	4	5	6	6	6	6	7	8	8
$N_K=1, N_G=2$	0	0	0	0	0	0	1	2	3	4	4	5	6	6	7	7	8	8	9	9	10	10	11	11
$N_K=2, N_G=1$	0	0	0	0	0	0	2	2	2	2	2	4	4	4	4	4	6	6	6	6	6	8	8	8
$N_K=2, N_G=2$	0	0	0	0	0	0	2	4	4	4	4	6	8	8	8	8	10	12	12	12	12	14	16	16
$N_K=3, N_G=1$	0	0	0	0	0	0	2	2	2	2	2	4	4	4	4	4	6	6	6	6	6	8	8	8
$N_K=3, N_G=2$	0	0	0	0	0	0	3	4	4	4	4	7	8	8	8	8	11	12	12	12	12	15	16	16

and internal tracks ($N_W = 8$ and $N_E = 8$) was kept fixed. The results of these experiments are illustrated in Figure 9.3 and summarized in Table 9.1 for clarity.

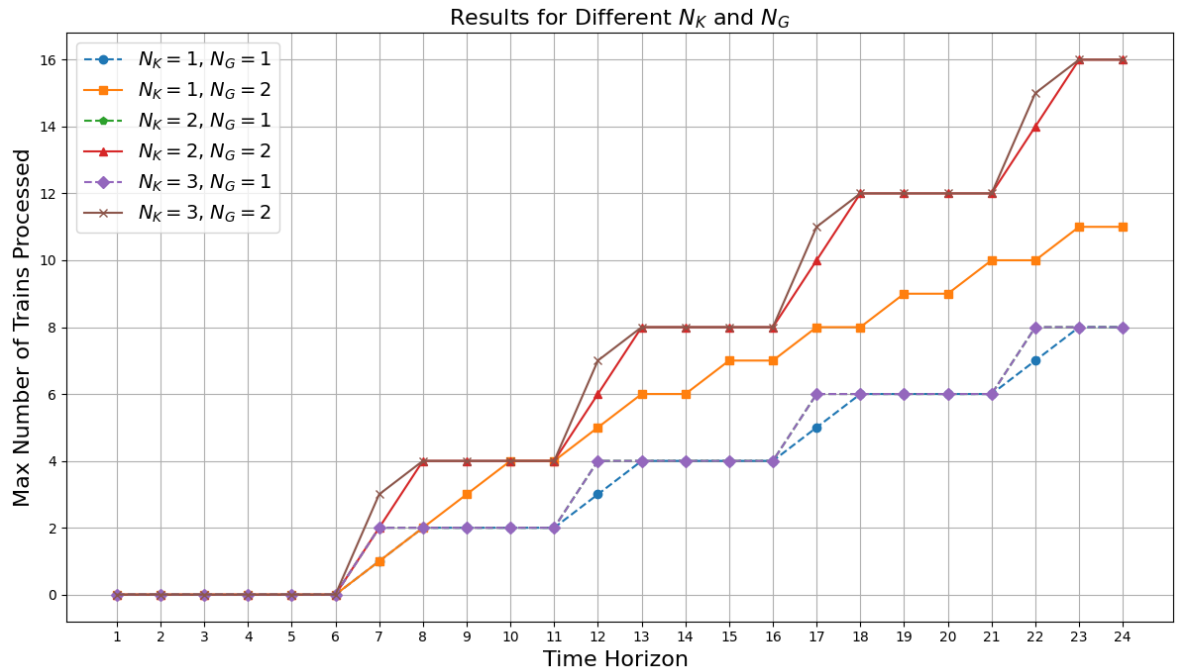


Figure 9.3: Number of trains processed for each (N_K, N_G) pair, $N_O = 5$

From the results, it can be observed that the number of completed trains increases when two intermediate tracks are available instead of only one, while keeping the number of cranes constant. This effect can be observed by comparing the blue and green curves, as well as the orange and red curves. The improvement occurs because, when two trains (for instance, one from the East and one from the West) complete their operations at the same time, both trains can depart simultaneously when two intermediate tracks are available.

However, introducing a third intermediate track does not lead to additional improvements when

only one RMG crane operates in each park. Under this configuration, the cranes become the limiting factor in the system. When two RMG cranes are used, improvements can be observed at certain time intervals (e.g., at $t = 17$ and $t = 22$), where a higher number of trains is completed.

It should be noted that although differences are observed at intermediate time steps, the total number of trains completed by the end of the 24 time intervals remains unchanged in some configurations. In particular, when only one RMG crane (N_G) is used, the final throughput remains the same regardless of the number of intermediate tracks (N_K).

What changes instead is the processing speed, which is reflected in the steeper slopes of the curves shown in the figure. Furthermore, increasing the number of RMG cranes results in a higher number of trains being processed overall, indicating that this parameter has the most significant impact on train throughput.

The optimization problem was solved for all combinations of 24 time steps, three different values of N_K , and two different values of N_G , resulting in a total of 144 scenarios. The total computational time required to solve all scenarios was approximately 6084 seconds.

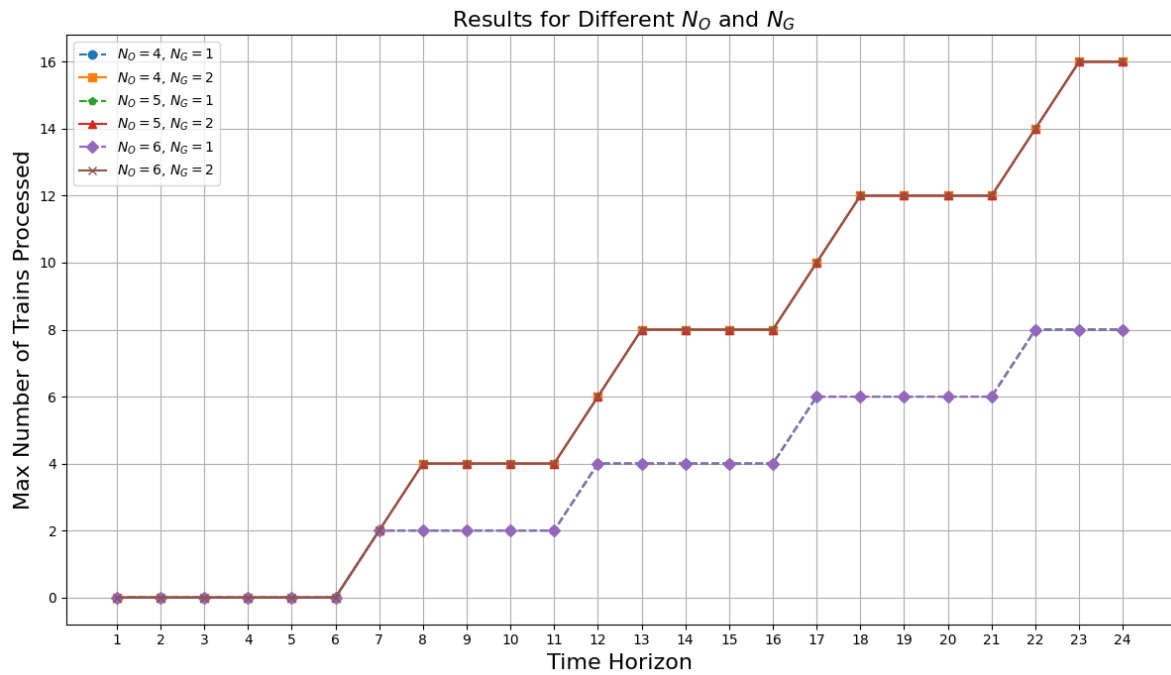


Figure 9.4: Number of trains processed for each (N_O, N_G) pair, $N_K=2$

Another experiment involves the variation of the number of external tracks (N_O), while keeping the number of internal tracks (N_W and N_E) fixed at 8, and the number of intermediate tracks constant at $N_K = 2$. As shown in Fig. 9.4, the plots for $N_O = 4$, $N_O = 5$ and $N_O = 6$, for the same number of cranes (N_G), are identical, indicating that increasing the number of external tracks

does not lead to any improvement in train throughput. The results remained unchanged also with $N_K = 3$, suggesting that the number of external tracks does not significantly impact performance in this configuration.

9.1.3 Process modeling with locomotive constraints

The model described in the previous section does not account for locomotive availability or for the number of locomotives as parameters of the system. In that formulation, it is implicitly assumed that a locomotive is always available whenever a train movement is required. To address this limitation, additional constraints are introduced to **Problem 7** in order to explicitly incorporate locomotive availability and evaluate its impact on the system performance.

Before presenting these constraints, the additional decision variables and parameters required to model locomotive operations are introduced.

- N_L : a parameter representing the maximum number of locomotives available;
- $l_{i,j}(t)$: a variable which is 1 if a locomotive is assigned to a train to make it enter from outside track i to the inside track j and 0 otherwise;
- $l'_{i,j}(t)$: a variable which is 1 if a locomotive is assigned to a train to make it leave the inside track j and go to outside track i after finished and 0 otherwise;
- $loco_{i,j}(t)$, $locoO_{i,j}(t)$, $locoI_{i,j}(t)$, $locoR_{i,j}(t)$: variables helping to linearize the products between other variables.

9.1.3.1 Constraints with locomotive considerations

The model to optimise now becomes:

Problem 8

$$\max \sum_{i=1}^{N_O} \sum_{j=1}^{N_I} \sum_{k=1}^{N_K} \sum_{t=1}^T y'_{i,j,k}(t) \quad (9.42)$$

subject to

(9.2)–(9.4), (9.6), (9.9), (9.10), (9.14)–(9.22), (9.24)–(9.41) and:

$$\begin{aligned} y_{i,j,k}(t+1) &\leq q_i^O(t) * l_{i,j}(t) \quad \forall i = 1, \dots, N_O \\ &\forall j = 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad \forall t = 1, \dots, T-1 \end{aligned} \quad (9.43)$$

$$\sum_{k=1}^{N_K} y_{i,j,k}(t+1) \geq \delta_{i,j}(t) * l_{i,j}(t) \quad \forall i = 1, \dots, N_O \quad (9.44)$$

$$\forall j = 1, \dots, N_I \quad \forall t = 1, \dots, T-1$$

$$y'_{i,j,k}(t) \leq q_j^W(t-1) * l'_{i,j}(t) \quad \forall i = 1, \dots, N_O \quad (9.45)$$

$$\forall j = 1, \dots, N_W \quad \forall k = 1, \dots, N_K \quad \forall t = 2, \dots, T$$

$$y'_{i,j,k}(t) \leq q_{j-N_W}^E(t-1) * l'_{i,j}(t) \quad \forall i = 1, \dots, N_O \quad (9.46)$$

$$\forall j = N_W + 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad \forall t = 2, \dots, T$$

$$y'_{i,j,k}(t+1) \leq r_j(t) * l'_{i,j}(t) \quad \forall i = 1, \dots, N_O \quad (9.47)$$

$$\forall j = 1, \dots, N_I \quad \forall k = 1, \dots, N_K \quad \forall t = 1, \dots, T-1$$

$$\sum_{i=1}^{N_O} \sum_{j=1}^{N_I} (l_{i,j}(t) + l'_{i,j}(t)) \leq N_L \quad \forall t = 1, \dots, T \quad (9.48)$$

$$l_{i,j}(1) \leq N_L \quad \forall i = 1, \dots, N_O, \quad \forall j = 1, \dots, N_I \quad (9.49)$$

$$l'_{i,j}(1) \leq N_L \quad \forall i = 1, \dots, N_O, \quad \forall j = 1, \dots, N_I \quad (9.50)$$

$$\sum_{i=1}^{N_O} \sum_{j=1}^{N_I} (l_{i,j}(t) + l_{i,j}(t-1)) \leq N_L \quad \forall t = 2, \dots, T \quad (9.51)$$

$$\sum_{i=1}^{N_O} \sum_{j=1}^{N_I} (l'_{i,j}(t) + l'_{i,j}(t-1)) \leq N_L \quad \forall t = 2, \dots, T \quad (9.52)$$

Constraints 9.43, 9.44, 9.45, 9.46 and 9.47 are nonlinear. In order to obtain a linear formulation, these constraints are linearized

$$q_i^O(t) + l_{i,j}(t) \leq 1 + locoO_{i,j}(t) \quad (9.53)$$

$$locoO_{i,j}(t) \leq q_i^O(t) \quad (9.54)$$

$$locoO_{i,j}(t) \leq l_{i,j}(t) \quad (9.55)$$

$$\forall i = 1, \dots, N_O, \forall j = 1, \dots, N_I, \forall t = 1, \dots, T$$

$$y_{i,j,k}(t) \leq locoO_{i,j}(t-1) \quad (9.56)$$

$$\forall i = 1, \dots, N_O, \forall j = 1, \dots, N_I, \forall k = 1, \dots, N_K,$$

$$\forall t = 2, \dots, T$$

$$\delta_{i,j}(t-1) + l_{i,j}(t-1) \leq 1 + loco_{i,j}(t-1) \quad (9.57)$$

$$\forall i = 1, \dots, N_O, \forall j = 1, \dots, N_I, \forall t = 2, \dots, T$$

$$loco_{i,j}(t) \leq \delta_{i,j}(t) \quad (9.58)$$

$$loco_{i,j}(t) \leq l_{i,j}(t) \quad (9.59)$$

$$\forall i = 1, \dots, N_O, \forall j = 1, \dots, N_I, \forall t = 1, \dots, T$$

$$\sum_{k=1}^{N_K} y_{i,j,k}(t) \geq loco_{i,j}(t-1) \quad (9.60)$$

$$\forall i = 1, \dots, N_O, \forall j = 1, \dots, N_I, \forall t = 2, \dots, T$$

$$q_j^W(t-1) + l'_{i,j}(t-1) \leq 1 + locoI_{i,j}(t-1) \quad (9.61)$$

$$locoI_{i,j}(t-1) \leq q_j^W(t-1) \quad (9.62)$$

$$locoI_{i,j}(t-1) \leq l'_{i,j}(t-1) \quad (9.63)$$

$$y'_{i,j,k}(t) \leq locoI_{i,j}(t-1) \quad (9.64)$$

$$\forall i = 1, \dots, N_O, \quad \forall j = 1, \dots, N_W,$$

$$\forall k = 1, \dots, N_K, \quad \forall t = 2, \dots, T$$

$$q_{j-N_W}^E(t-1) + l'_{i,j}(t-1) \leq 1 + locoI_{i,j}(t-1) \quad (9.65)$$

$$locoI_{i,j}(t-1) \leq q_{j-N_W}^E(t-1) \quad (9.66)$$

$$locoI_{i,j}(t-1) \leq l'_{i,j}(t-1) \quad (9.67)$$

$$y'_{i,j,k}(t) \leq locoI_{i,j}(t-1) \quad (9.68)$$

$$\forall i = 1, \dots, N_O, \quad \forall j = N_W+1, \dots, N_I,$$

$$\forall k = 1, \dots, N_K, \quad \forall t = 2, \dots, T$$

$$r_j(t) + l'_{i,j}(t) \leq 1 + locoR_{i,j}(t) \quad (9.69)$$

$$locoR_{i,j}(t) \leq r_j(t) \quad (9.70)$$

$$locoR_{i,j}(t) \leq l'_{i,j}(t) \quad (9.71)$$

$$\forall i = 1, \dots, N_O, \quad \forall j = 1, \dots, N_I, \quad \forall t = 1, \dots, T$$

$$y'_{i,j,k}(t+1) \leq locoR_{i,j}(t) \quad (9.72)$$

$$\forall i = 1, \dots, N_O, \quad \forall j = 1, \dots, N_I,$$

$$\forall k = 1, \dots, N_K, \quad \forall t = 1, \dots, T-1$$

The nonlinear constraints presented above involve the product of binary variables. To obtain linear formulation, these terms are linearized using a standard reformulation for the product of two binary variables.

For example, let $x, y \in \{0, 1\}$ and let z be a binary variable such that

$$z = x \cdot y.$$

This relationship can be equivalently represented by the following set of linear constraints:

$$x + y \leq 1 + z \quad (9.73)$$

$$z \leq x \quad (9.74)$$

$$z \leq y \quad (9.75)$$

This formulation ensures that z takes value 1 if and only if both x and y are equal to 1, and 0 otherwise.

This implies that Constraint 9.43 is linearized through Constraints 9.53–9.56, Constraint 9.44 through Constraints 9.57–9.60, Constraint 9.45 through Constraints 9.61–9.64, Constraint 9.46 through Constraints 9.65–9.68, and Constraint 9.47 through Constraints 9.69–9.72.

Constraint 9.43 ensures that a train can enter the terminal only if it was previously outside and if a locomotive is available and assigned to that train.

Constraint 9.44 guarantees that trains waiting on the external tracks are given priority to enter the terminal at time t over newly arriving trains at the same time. Moreover, it ensures that a train can enter only if a locomotive has been assigned to it.

Constraints 9.45 and 9.46 ensure that a train can exit the terminal only if it was previously located on an internal track and if a locomotive is available and assigned to the train.

Constraint 9.47 guarantees that a train can leave an internal track only after completing its operations and only if a locomotive is available.

Constraint 9.48 ensures that the total number of locomotives assigned to entrance and exit operations does not exceed the total number of locomotives available in the system, denoted by L .

Constraints 9.49 and 9.50 define the initial conditions for locomotive assignments.

Finally, Constraints 9.51 and 9.52 ensure that a locomotive cannot be used in two consecutive time periods, since a locomotive requires one time step to return and retrieve another train.

9.1.3.2 Results with locomotive constraints

When locomotives are explicitly introduced as a constraint in the model, the experiment focuses on analysing the impact of varying both the number of locomotives and the number of RMG cranes, while keeping all other parameters fixed according to the current configuration of the real system ($N_O = 5$, $N_K = 2$, $N_W = 8$, $N_E = 8$). The number of locomotives is varied between 1 and 2, as these values realistically represent the typical availability in practice.

The results of the optimization problem for different combinations of locomotives and RMG cranes are presented in Figure 9.5. Even under the considered theoretical assumptions, namely that each train requires 1 hour to enter, 1 hour to exit, and 5 hours to be fully processed, the RMG cranes remain a key factor influencing the overall system performance. A more detailed analysis shows that, when the number of locomotives is kept fixed, increasing the number of gantry cranes from $N_G = 1$ to $N_G = 2$ leads to a substantial increase in the maximum number of processed trains across all time horizons. This improvement is consistently observed for both $N_L = 1$ and $N_L = 2$, indicating that crane capacity plays a major role in enhancing system throughput. In contrast, when the number of gantry cranes is fixed, increasing the number of locomotives from $N_L = 1$ to $N_L = 2$ results in a more moderate improvement. While additional locomotives

do contribute to better performance, the magnitude of this effect is noticeably smaller than that achieved by increasing crane capacity. This asymmetric behaviour suggests that, under finite locomotive availability, the system is more sensitive to variations in the number of gantry cranes than to changes in the number of locomotives. Consequently, gantry cranes can be identified as the dominant resource in determining system performance in these configurations.

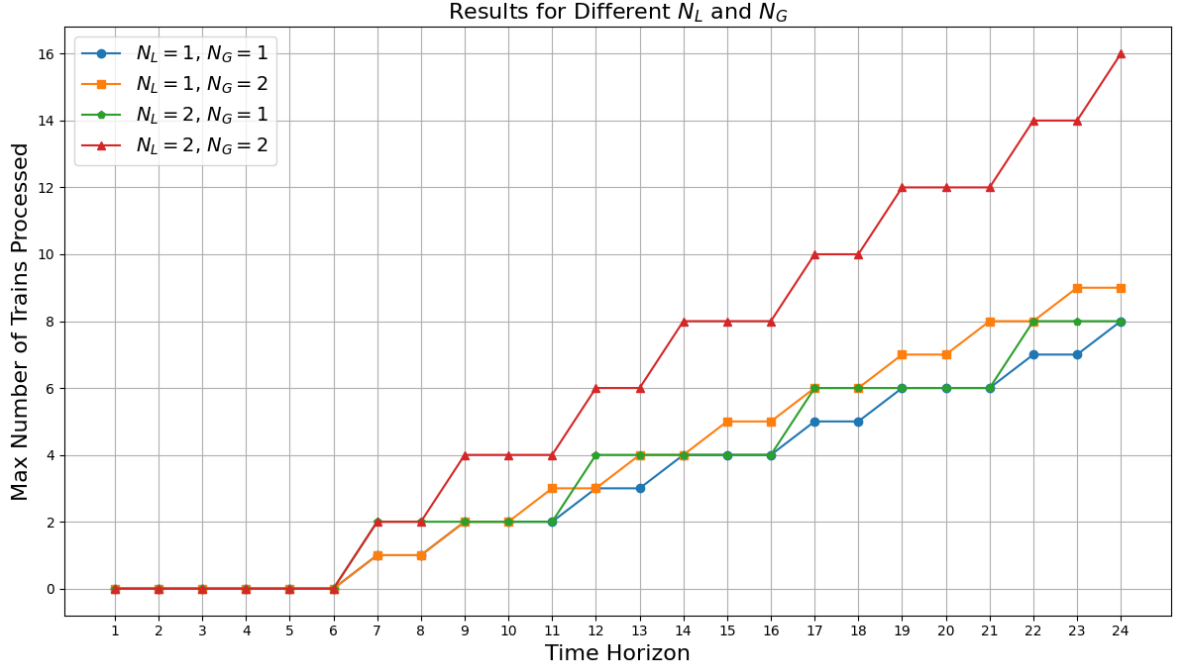


Figure 9.5: Number of trains processed for each (N_L, N_G) pair, $N_O = 5, N_K = 2$

This interpretation becomes more insightful when comparing the finite locomotive scenarios with the case of unconstrained locomotive availability ($N_L = \infty$) shown in Fig. 9.6. When the number of gantry cranes is fixed at $N_G = 1$, increasing the number of locomotives, even to infinity, does not lead to any significant improvement in system performance. This confirms that, in this configuration, the gantry crane is the binding constraint, and additional locomotives remain underutilised. However, when the number of gantry cranes is increased to $N_G = 2$, the effect of locomotives becomes more pronounced. In this case, moving from a finite number of locomotives to an unbounded availability results in a clear increase in the number of processed trains, indicating that locomotives start to act as a limiting resource once sufficient crane capacity is provided. Nevertheless, the performance gain obtained by increasing N_L diminishes as the number of locomotives grows, suggesting that beyond a certain point, additional locomotives yield limited benefits. Overall, these results demonstrate that the system bottleneck depends on which parameter is fixed: when crane capacity is low, it dominates system performance, whereas when crane capacity is increased, the limitation shifts towards locomotive availability.

For better readability, the results are also presented in Table 9.2, where the last two rows, highlighted in blue, correspond to the results obtained from the previous model in which no constraints on the number of locomotives were considered.

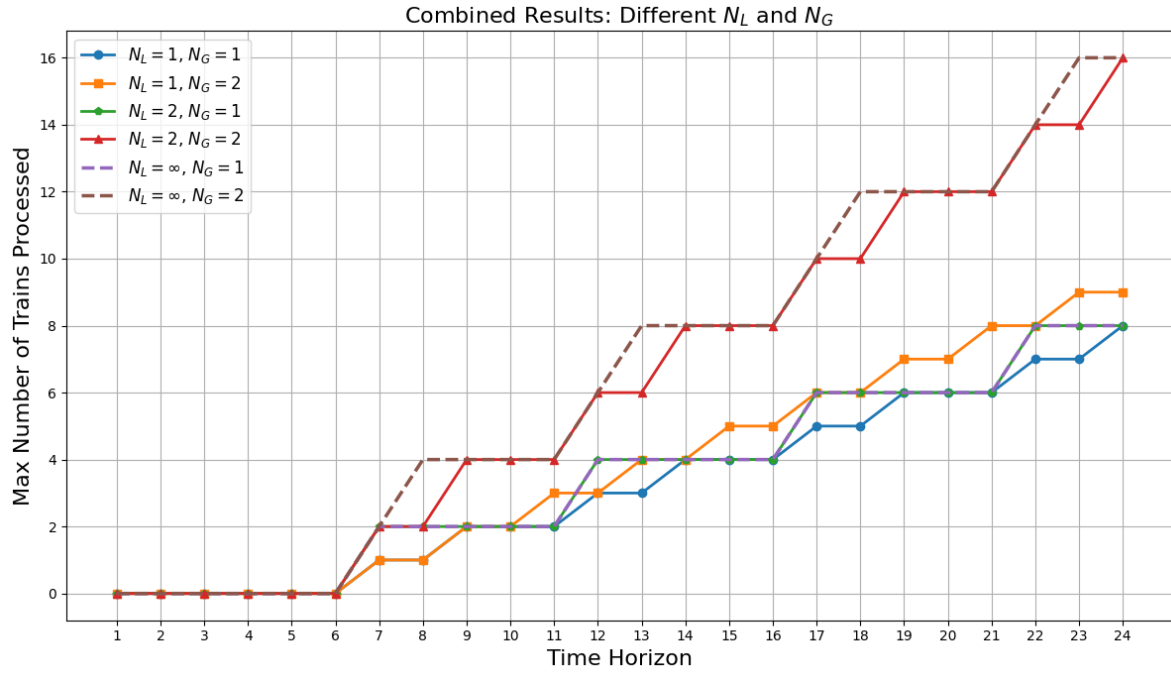


Figure 9.6: Number of trains processed for each (N_L, N_G) pair, $N_O = 5, N_K = 2$

Table 9.2: Maximum number of trains processed for different values of N_L and N_G . The last two rows correspond to the dashed curves, i.e. the cases with unconstrained locomotives ($N_L = \infty$).

(T)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$N_L=1, N_G=1$	0	0	0	0	0	0	1	1	2	2	2	3	3	4	4	4	5	5	6	6	6	7	7	8
$N_L=1, N_G=2$	0	0	0	0	0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9
$N_L=2, N_G=1$	0	0	0	0	0	0	2	2	2	2	2	4	4	4	4	4	6	6	6	6	6	8	8	8
$N_L=2, N_G=2$	0	0	0	0	0	0	2	2	4	4	4	6	6	8	8	8	10	10	12	12	12	14	14	16
$N_L=\infty, N_G=1$	0	0	0	0	0	0	2	2	2	2	2	4	4	4	4	4	6	6	6	6	6	8	8	8
$N_L=\infty, N_G=2$	0	0	0	0	0	0	2	4	4	4	4	6	8	8	8	8	10	12	12	12	12	14	16	16

9.2 Capacity evaluation through a simulation model

In the previous section (Sec.9.1), the capacity of the rail yard was analyzed using a mathematical programming approach. While this method provides a structured analytical estimation of system

capacity, it does not fully capture the dynamic behavior of real rail-yard operations. In practice, rail-yard processes involve time-dependent interactions among train arrivals, track availability, resource allocation, and handling operations. Additionally, operational factors such as realistic train entry and departure times, variability in handling durations, and priority rules for train or resource allocation may significantly influence the effective capacity of the rail yard.

For this reason, a discrete-event simulation model was developed in FlexTerm to reproduce the operational behavior of the rail yard under dynamic conditions. Simulation enables the representation of stochastic processes, queue formation, and competition for shared resources, thus providing a more realistic evaluation of system performance.

The simulation model is therefore used as a complementary tool to the mathematical formulation, allowing a deeper analysis of the operational capacity of the rail yard and facilitating the identification of potential bottlenecks under different operating scenarios.

In order to systematically analyze the behavior of the rail-yard system, the simulation experiments were organized in three progressive phases.

First, the theoretical configuration derived from the mathematical programming model was reproduced within the simulation environment. This step allows the comparison between analytical capacity estimations and the results obtained from a dynamic simulation model.

Second, additional operational elements were introduced in order to represent more realistic rail-yard conditions. In particular, additional operational logic was implemented to account for locomotive availability and its impact on train movements within the yard.

Finally, stress-test experiments were conducted by introducing a large number of trains simultaneously into the system. These experiments allow the evaluation of the maximum operational capacity of the rail yard and the identification of potential bottlenecks when the system operates under extreme demand conditions.

9.2.1 Structure and logic of the capacity model in FlexTerm

In this section, the main components of the simulation model and the algorithms governing their behavior are presented in order to represent the train arrival, handling, and departure processes within the terminal.

The subsection is organized as follows. First, the physical components of the simulation model are described. Subsequently, the operational logic governing the interactions among the system elements is presented.

9.2.1.1 Model components

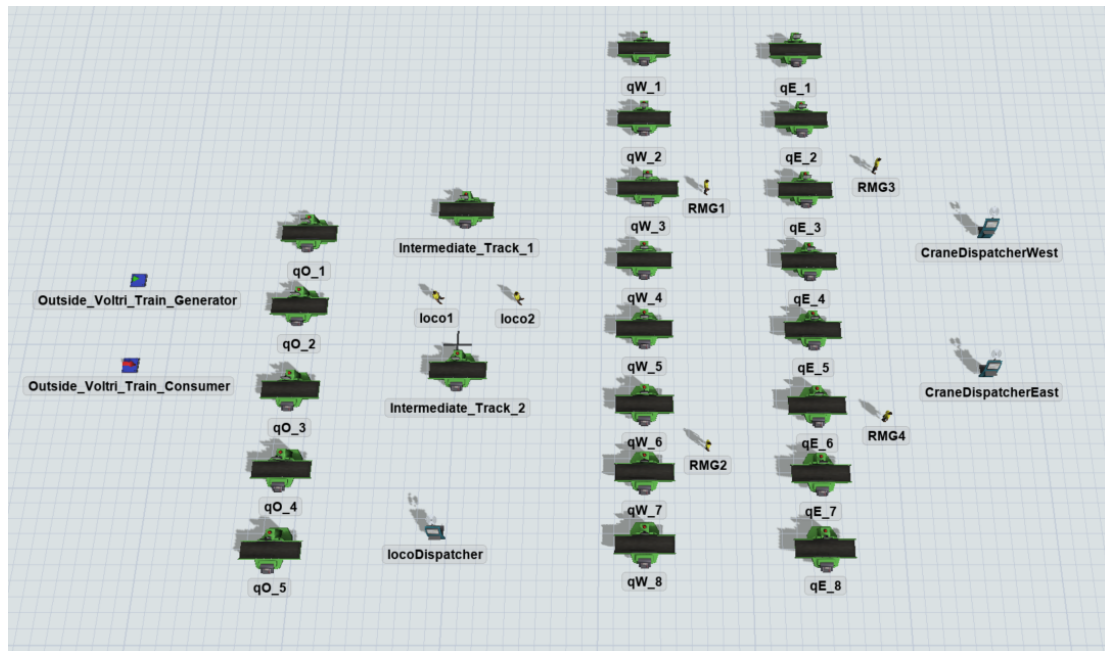


Figure 9.7: Capacity simulation model in Flex Term

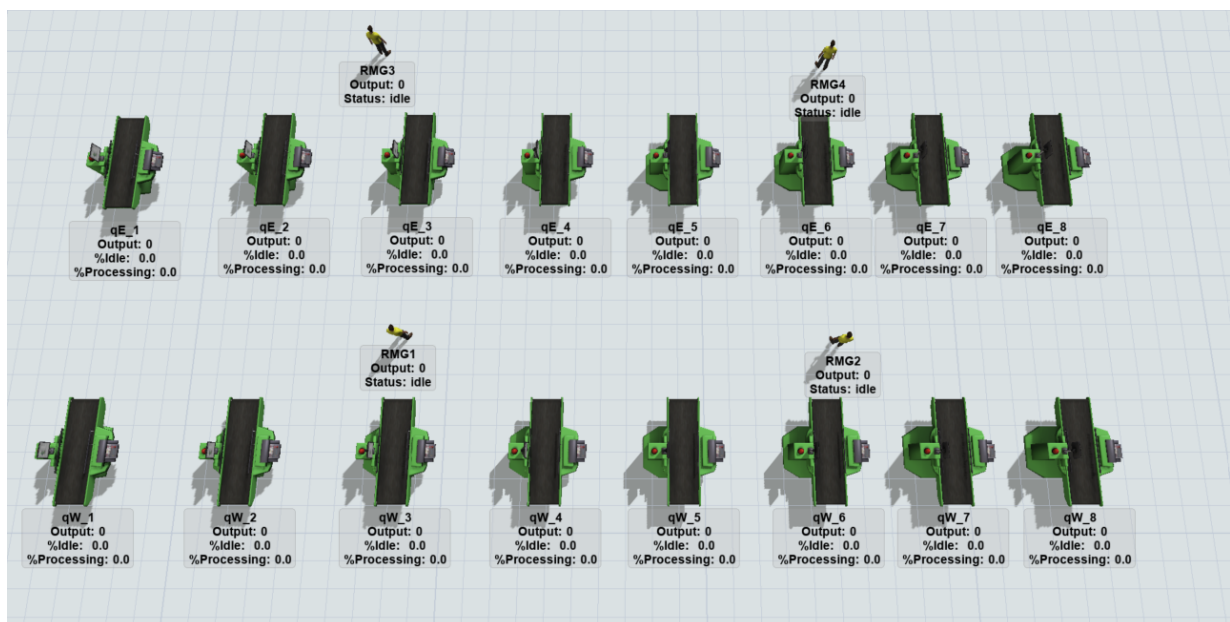


Figure 9.8: Internal rail park

The simulation model of the PSA GP rail yard was constructed by mapping each element of the physical system to FlexTerm objects as follows:

- **Trains:** Represented by entities generated at a *Source*, which specifies arrival times and basic characteristics such as processing time distributions and container loads.
- **External Tracks:** Modeled as processors that hold trains upon arrival. The release of trains from the external tracks is subject to the availability of locomotives, intermediate tracks, and suitable entry conditions for the internal tracks.
- **Intermediate Tracks:** Implemented as processors responsible for transferring trains between the external and internal tracks.
- **Internal Tracks (West/East):** Modeled as processors where trains undergo container handling.
- **Locomotives:** Represented by operators whose availability and dispatching are controlled by dispatcher objects. Locomotives are responsible for moving trains between tracks and are subject to stochastic availability constraints.
- **RMG Cranes:** Modeled as additional operators assigned by RMG dispatchers. Each train requests an RMG once on an internal track; dispatching of RMGs follows a priority rule decided by us.
- **Dispatchers:** FlexTerm dispatcher objects are used to manage locomotive assignments to intermediate tracks and to coordinate RMG allocation across the West and East parks.
- **Sink:** A terminal sink object represents train departure from outside tracks once processing and exit conditions are fulfilled.

Fig. 9.7 provides a schematic view of the simulation model, illustrating its main components. Meanwhile, Fig. 9.8 presents a more detailed view of the internal rail yard, focusing on the 16 tracks of the terminal's internal rail park and showing the corresponding RMG cranes.

9.2.1.2 Connections between model components

In the simulation model, the different components are connected in order to represent the flow of trains through the terminal. These connections define how trains move between the various elements of the system, including the source, outside tracks, intermediate tracks, inside tracks, and the sink.

The main elements used to establish these connections are the ports associated with each object. In FlexSim and FlexTerm, a port represents a point on an object through which flow items can be

transferred to another object. A port connection therefore defines the relationship between two objects, allowing them to exchange flow items and communicate with each other.

One of the main advantages of using port connections is that they allow the model structure to be created in a flexible and intuitive way. Different types of port connections are available in FlexSim and FlexTerm, and several of these are used in the simulation model.

Input/Output Ports (A-Connections)

Input/output ports, also referred to as A-connections, are the most common type of port connection. These ports are typically used to connect two fixed resources so that they can exchange flow items. In this case, the output port of an upstream object is connected to the input port of a downstream object. The output port represents the point where a flow item leaves an object, while the input port represents the point where it enters the next object.

In the simulation model, this type of connection is used between the source and the outside tracks, between the outside tracks and the sink, between the outside tracks and the intermediate tracks (and vice versa), and between the intermediate tracks and the inside tracks (and vice versa). In addition, A-connections are used between the locomotive dispatcher and the locomotives, as well as between each RMG dispatcher and the corresponding RMG cranes.

Center Ports (S-Connections)

Another type of connection used in the model is the center port connection, also known as S-connection. Center ports are typically used to connect task executors, such as dispatchers and operators modeling locomotives and RMG cranes, to fixed resources such as processors modeling the tracks. However, center ports can also be used more generally to allow two objects to reference each other.

When the center ports of two objects are connected, an abstract reference relationship is created between them. This allows objects to communicate and interact in several ways. For example, fixed resources may use the task executors connected to their center ports to transport flow items to other resources. Similarly, processors with setup or processing times may require the presence of a task executor connected through a center port in order to perform their operations. Center ports can also be used simply to allow objects to exchange information.

In the simulation model, center port connections are used to link the locomotive dispatcher with all track elements, including the outside tracks, intermediate tracks, and inside tracks.

9.2.1.3 Operational logic and rules

The simulation logic reflects the main operational constraints and decision rules observed at the terminal.

All operational rules were implemented in the *FlexTerm* environment using custom *FlexScript*

code to define decision logic, dispatching rules, and stochastic behavior of resources. This simulation framework enables the evaluation of locomotive constraints on system capacity and supports testing of alternative operational policies and resource configurations.

Operational logic in source:

In the simulation model, the Source object does not immediately release a train into the system. Instead, when a train becomes available at the source, the Source sends a message to its Message Trigger, which activates a control mechanism to determine whether the train can be safely introduced into the rail yard. The purpose of this control logic is to verify the availability of a feasible inside track before releasing the train, thereby preventing blocking situations and deadlocks within the model.

Depending on the trigger state, different actions are performed. If the train is already confirmed as ready for release, it is directly routed toward the selected track. Otherwise, the logic checks the availability of an internal destination according to one of three possible allocation policies: random inside-track assignment, assignment to the planned track, or assignment to a random track within the least busy park. If no feasible inside track is found, the train is not released and remains at the source until the control logic is called again after a delay.

This mechanism ensures that trains enter the rail yard only when a valid continuation path is available. As a result, the release policy improves the robustness of the simulation model and prevents deadlocks caused by premature train entry.

The control logic used to release trains from the source is summarized in Algorithm 5.

Algorithm 5 Train release logic from the source

Require: Train waiting at the source, trigger type, track-selection policy

Ensure: Train is released to the rail yard only if a feasible inside track exists

```
1: Retrieve the train currently waiting at the source
2: if no train exists then
3:   Stop procedure
4: end if
5: Read the trigger type associated with the message
6: if trigger = TRAIN_READY then
7:   Release the train toward the chosen track
8: end if
9: if trigger = FIRST_CALL_OR_AFTER_DELAY then
10:  Read the inside-track selection policy
11:  if policy = random inside track then
12:    Check if a random inside track is available
13:  else if policy = planned track then
14:    Check if the planned inside track is available
15:  else if policy = least busy park then
16:    Check if a random inside track in the least busy park is available
17:  end if
18:  if a feasible inside track is found then
19:    Release the train toward the selected track
20:  else
21:    Keep the train at the source and repeat the check after a delay
22:  end if
23: end if
24: if trigger = AFTER_DELAY_RANDOM then
25:  Check again if a random inside track is available
26: end if
27: if trigger = AFTER_DELAY_RANDOM_BEST_PARK then
28:  Check again if a random inside track in the least busy park is available
29: end if
30: if trigger = AFTER_DELAY_PLANNED_TRACK then
31:  Check again if the planned inside track is available
32: end if
33: if a feasible inside track is available then
34:  Release the train
35: else
36:  Keep the train at the source until the next trigger
37: end if
```

Among the available inside-track allocation policies, the random inside-track selection rule is the one most frequently used in the simulation experiments. For this reason, its operational logic is described in more detail below and summarized in Algorithm 6.

The random inside-track selection procedure is used to identify a feasible pair of destination resources before a train is released from the source. More specifically, the procedure first scans the internal tracks of the rail yard and collects all tracks that are currently available. In the West park, the search starts from the second track, so that the uppermost West track is intentionally kept free for train passage to/from the east tracks.

An inside track is considered feasible only if it is neither busy nor reserved. In addition, for the West park, tracks reserved for passing movements toward or from the East park are also excluded from the candidate set. The same feasibility check is then performed for the outside tracks, where only tracks that are neither busy nor reserved are considered available.

If at least one inside track and at least one outside track are simultaneously available, one inside track is selected randomly from the set of feasible internal tracks and assigned to the train. The selected inside track is then marked as reserved in order to avoid conflicting assignments. For experimental analysis, the model also records whether the chosen inside track belongs to the East or West park.

Subsequently, one outside track is selected randomly from the set of feasible outside tracks. The train is then not released directly; instead, a message is sent to the source control logic so that the train can be released through the port corresponding to the selected outside track.

If either no feasible inside track or no feasible outside track is available, the train is not released. In this case, a delayed message is generated so that the source control logic is called again after a fixed delay. The allocation method to be used after the delay may vary according to the selected model parameter, allowing the train to be re-evaluated using random selection, selection within the least busy park, or assignment to the planned track.

This logic ensures that trains are released only when both an internal destination and an external receiving track are available, thereby reducing blocking situations and preventing deadlocks in the simulation model.

Algorithm 6 Random selection of inside and outside tracks

Require: Train waiting at the source

Ensure: Train is released only if both a feasible inside track and a feasible outside track are available

```
1: Initialize the set of feasible inside tracks as empty
2: Initialize the set of feasible outside tracks as empty
3: for each track in the West park, starting from the second track do
4:   if the track is not busy, not reserved, and not reserved for passing movements then
5:     Add the track to the set of feasible inside tracks
6:   end if
7: end for
8: for each track in the East park do
9:   if the track is not busy and not reserved then
10:    Add the track to the set of feasible inside tracks
11:   end if
12: end for
13: for each track in the outside park do
14:   if the track is not busy and not reserved then
15:    Add the track to the set of feasible outside tracks
16:   end if
17: end for
18: if the set of feasible inside tracks is not empty and the set of feasible outside tracks is not empty then
19:   Randomly select one inside track from the feasible inside-track set
20:   Assign the selected inside track to the train
21:   Mark the selected inside track as reserved
22:   Randomly select one outside track from the feasible outside-track set
23:   Identify the output port corresponding to the selected outside track
24:   Send a message to the source control logic to release the train through the selected outside track
25: else
26:   Do not release the train
27:   Read the policy to be used after the delay
28:   if the delay policy is random selection then
29:     Send a delayed message to retry with random selection
30:   else if the delay policy is least busy park then
31:     Send a delayed message to retry with selection in the least busy park
32:   else if the delay policy is planned track then
33:     Send a delayed message to retry with the planned-track policy
34:   end if
35:   Increase the train delay accumulated at the source
36: end if
```


Operational logic of the outside tracks:

The outside tracks are modeled as processor objects representing the time a train spends waiting before entering the internal rail yard. The processing time associated with these processors represents the time required for a train to remain in the outside track and is defined through model parameters.

In addition to the processing time, entry and exit triggers are used to update several labels associated with the train and the tracks. These labels are used to track the state of the system, for instance whether a track is busy or reserved.

Once the train has completed its processing time in the outside track, it is not immediately released toward the internal tracks. Instead, a message is sent to the locomotive dispatcher, which acts as a central control mechanism for train movements within the rail yard. The dispatcher verifies whether the necessary conditions for moving the train are satisfied. In particular, it checks the availability of both an intermediate track and a locomotive that can perform the movement.

If these resources are available, the dispatcher assigns the locomotive to the train and sends a message back to the outside track authorizing the release of the train toward the selected intermediate track. Otherwise, the train remains in the outside track until the required resources become available.

Operational logic of the locomotive dispatcher:

The dispatcher acts as a coordination mechanism that ensures trains are allowed to move only when all the resources required for the next movement are simultaneously available.

Algorithm 7 summarizes the decision logic implemented in the locomotive dispatcher when a train is waiting in an outside track.

In particular, three main conditions must be satisfied before a train can be released: the availability of a feasible intermediate track, the presence of at least one operationally available locomotive, and, when the destination inside track belongs to the East park, the availability of the corresponding West crossing track. Only when these conditions are satisfied are the required resources reserved and the train authorized to move toward the selected intermediate track.

Locomotive availability is not deterministic in the model. Even when a locomotive is idle, it may not be operationally available. This behavior is modeled through a Bernoulli random variable with parameter p , representing the probability that a locomotive is available at the time of assignment.

Algorithm 7 Locomotive dispatcher logic for trains in outside tracks

Require: Message from an outside track requesting the movement of a train

Ensure: The train is released only if all resources required for the movement are available

```
1: Retrieve the train waiting in the outside track
2: if the train has already completed its cycle then
3:   Apply the corresponding exit logic
4:   Stop
5: end if
6: Identify a feasible intermediate track connected to the outside track
7: Select the feasible intermediate track with the lowest current occupation
8: Initialize the set of candidate locomotives
9: for each locomotive do
10:   if the locomotive is idle and not reserved then
11:     Sample  $A \sim \text{Bernoulli}(p)$ 
12:     if  $A = 1$  then
13:       Add the locomotive to the set of candidate locomotives
14:     else
15:       Schedule its future availability after a delay
16:     end if
17:   end if
18: end for
19: if the assigned inside track belongs to the East park then
20:   Check whether the corresponding West crossing track is available
21: end if
22: if a feasible intermediate track exists and at least one locomotive is operationally available
   and the West crossing track, when required, is available then
23:   Select one locomotive from the candidate set
24:   Reserve the selected locomotive
25:   Reserve the selected intermediate track
26:   if the destination inside track belongs to the East park then
27:     Reserve the corresponding West crossing track
28:   end if
29:   Send an authorization message to the outside track
30:   Release the train toward the selected intermediate track
31: else
32:   Send a delayed message to the dispatcher
33:   Repeat the evaluation after the predefined delay
34: end if
```

If one or more of the required conditions are not satisfied, the dispatcher postpones the movement by sending a delayed message and reevaluates the request after a predefined time interval. This mechanism prevents trains from leaving the outside tracks prematurely and reduces the risk of blocking situations within the rail yard.

The same dispatcher logic is applied when a train located in an inside track requests permission to leave the rail yard. In this case, the inside track sends a message to the locomotive dispatcher in order to verify whether the required resources for the outbound movement are available.

Meanwhile, when a train is located in an inside track and requests to leave the rail yard, the decision process is similar to the one previously described. However, an additional condition must be satisfied: in addition to the availability of a locomotive and the required intermediate track, the dispatcher must also verify that at least one outside track is available to receive the departing train. Only when all these conditions are satisfied does the dispatcher authorize the train movement and reserve the necessary resources.

If any of the required resources is not available, the request is delayed and the dispatcher reevaluates the movement after a predefined time interval.

Operational logic of the intermediate tracks:

Similar to the outside tracks, the intermediate tracks are modeled using processor objects. The logic associated with these objects primarily concerns the processing time, which represents the time required for a train to move through the intermediate track.

The processing time depends on the scenario considered. In some cases, it is modeled as a fixed value, while in others it is generated from a probability distribution. In particular, for the scenarios based on operational data, the processing time is sampled from a triangular distribution whose parameters are derived from the observed operation times.

When a train leaves the yard from an inside track toward an outside track, an additional operational consideration arises if the train originates from an East track. In reality, the train must physically pass through a West track before exiting the yard. Since this passage is not explicitly modeled as a separate movement in the simulation, the corresponding travel time is incorporated into the processing time of the intermediate track.

During this period, the locomotive and the West track used for the passage remain reserved for the duration of the movement. This ensures that the required resources remain occupied for the appropriate amount of time and prevents other trains from using them until the movement is completed.

In addition to the processing time, the intermediate tracks also control the routing of trains after the processing operation is completed. If the train is entering the rail yard, it is directed toward the previously assigned inside track. Conversely, if the train is leaving the yard, it is routed toward the previously selected outside track.

The processor object also uses several event triggers, including *On Entry*, *On Exit*, *On Finish*, and *On Message*. These triggers are used to update the labels associated with both trains and tracks. Such labels store information about the current status of the system, such as whether a track is busy or reserved, and are essential for implementing the operational logic of the model.

Operational logic of the inside tracks:

Similarly to the other track types, the inside tracks are represented as processor objects in the simulation model. However, the logic associated with the inside tracks is more complex, as these tracks represent the main area where train handling operations take place.

As in the other processors, several event triggers are implemented. The *On Entry* trigger is used to update the labels associated with both the train and the track. In addition, depending on the selected model parameters, this trigger also determines the type of train arrival input that will be used in the simulation.

The *On Exit*, *On Finish*, and *On Message* triggers are mainly used to update the status labels of the trains and tracks. These labels store information about the current system state and are required for implementing the operational logic of the model.

Setup Time: The setup time of the processor in FlexTerm represents the preparation time required before a train can start being processed on the inside track. In the model, this time is used to represent the waiting period before the handling operations begin.

The value of the setup time depends on the operational scenario considered. In some cases, trains may be processed immediately upon arrival, while in other scenarios they may need to wait until the beginning of the next working shift. In the more realistic configurations of the model, the setup time also includes additional operational activities, such as the time required to decouple the locomotive before the train handling operations start.

Processing Time: The processing time defines the duration required to complete the handling operations for a train. Different formulations are used depending on the scenario being analyzed.

In the theoretical scenarios, a fixed processing time is assumed for all trains, representing the average time required to load or discharge a train. In this case, all trains are treated as identical.

In the more realistic scenarios, the processing time depends on the number of containers handled by the train. The input data describing train types and the number of containers handled for loading and unloading are stored in tables within FlexTerm.

Since it was not possible to define a suitable probability distribution describing the number of containers handled by each train type, the number of containers is generated using an empirical distribution derived from historical operational data. Then, based on the terminal data regarding the time required to load or discharge a container, the processing time for each container is sampled from a triangular distribution.

The total processing time for a train is therefore obtained by combining the handling times of all containers associated with that train. The procedure used to compute this value is described in the following algorithm.

Algorithm 8 Computation of train processing time on inside tracks

Require: train item, scenario type *InputType*

Ensure: processing time T

```

1: if InputType = THEORETICAL then
2:    $T \leftarrow 18000$ 
3:   item.HandlingTime  $\leftarrow T$ 
4:   return  $T$ 
5: end if
6: Retrieve the train type trainName from the corresponding arrival table
7: if trainName not contained in the train distribution table then
8:    $T \leftarrow 18000$ 
9:   item.HandlingTime  $\leftarrow T$ 
10:  return  $T$ 
11: end if
12:  $T_{\text{dis}} \leftarrow 0$ 
13:  $T_{\text{load}} \leftarrow 0$ 
14: if distribution data for IN containers exist then
15:    $N_{\text{IN}} \leftarrow \text{Empirical}(D_{\text{IN}})$ 
16:   for  $i = 1$  to  $N_{\text{IN}}$  do
17:      $t_i^{\text{dis}} \sim \text{Triangular}(0.8\theta_{\text{dis}}, \theta_{\text{dis}}, 1.2\theta_{\text{dis}})$ 
18:      $T_{\text{dis}} \leftarrow T_{\text{dis}} + t_i^{\text{dis}}$ 
19:   end for
20: end if
21: if distribution data for OUT containers exist then
22:    $N_{\text{OUT}} \leftarrow \text{Empirical}(D_{\text{OUT}})$ 
23:   for  $j = 1$  to  $N_{\text{OUT}}$  do
24:      $t_j^{\text{load}} \sim \text{Triangular}(0.8\theta_{\text{load}}, \theta_{\text{load}}, 1.2\theta_{\text{load}})$ 
25:      $T_{\text{load}} \leftarrow T_{\text{load}} + t_j^{\text{load}}$ 
26:   end for
27: end if
28:  $T \leftarrow T_{\text{dis}} + T_{\text{load}} + T_{\text{add}}$ 
29: item.HandlingTime  $\leftarrow T$ 
30: return  $T$ 

```

The variables used in Algorithm 8 are defined as follows:

- *InputType*: scenario defining the type of train arrivals considered in the simulation. Possible values are THEORETICAL, REALISTIC, and REALISTIC_FOR_SATURATION.
- T : total processing time of the train.
- T_{dis} : total time required for discharging containers from the train.
- T_{load} : total time required for loading containers onto the train.
- T_{add} : additional operational time included in the processing time (e.g., preparation or operational delays).
- N_{IN} : number of containers to be discharged from the train.
- N_{OUT} : number of containers to be loaded onto the train.
- t_i^{dis} : discharge time of container i .
- t_j^{load} : loading time of container j .
- θ_{dis} : average handling time required to discharge a container.
- θ_{load} : average handling time required to load a container.
- D_{IN} : table containing the empirical distribution of the number of containers to be discharged.
- D_{OUT} : table containing the empirical distribution of the number of containers to be loaded.
- $\text{Empirical}(D)$: function that samples a value from the empirical distribution defined by table D .

RMG request and release

Another component of the inside-track logic concerns the use of operators responsible for handling trains. In the simulation model, in this case, these operators correspond to RMG cranes. When a train arrives at an inside track and is ready to be processed, the processor requests an operator from the corresponding RMG dispatcher.

The dispatcher is responsible for selecting an available RMG and assigning it to the train. The detailed logic governing this selection process is described in the section dedicated to the RMG dispatcher.

Once the handling operations are completed, the inside-track processor releases the operator through a specific function, making the RMG available again for subsequent train-handling operations.

Analysis on train arrivals and departures

Here, two representative examples of the data related to train operations used as input for the simulation model are presented, together with the motivations for adopting an empirical distribution to describe them. For each train type, the empirical distribution of the number of containers is compared with commonly used theoretical distributions, such as the normal and triangular distributions. As illustrated in the selected examples (Figs. 9.9) and 9.10), although these theoretical distributions are able to capture the general trend of the data, they fail to accurately represent its discrete nature and variability, particularly in the tails and around the most frequent values.

For other train types, the fitting of standard parametric distributions proved to be even more challenging, with larger discrepancies observed between the empirical data and the theoretical approximations. In order to avoid introducing modelling inaccuracies, the empirical distribution was therefore selected for all train types, as it allows the real variability of the data to be preserved without imposing restrictive assumptions. For completeness, the full set of distribution analyses for all considered train types is reported in Appendix 10.

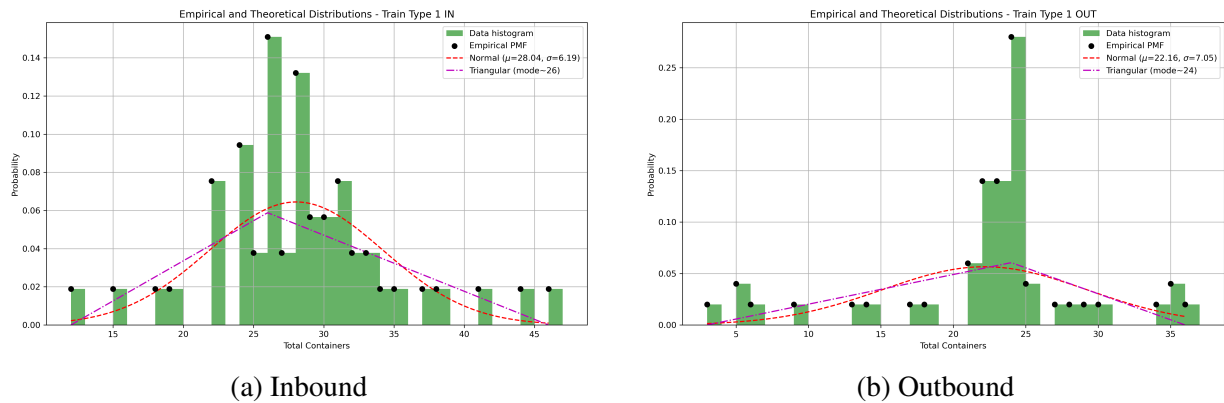


Figure 9.9: Distribution of containers for Train Type 1: (a) inbound trains and (b) outbound trains.

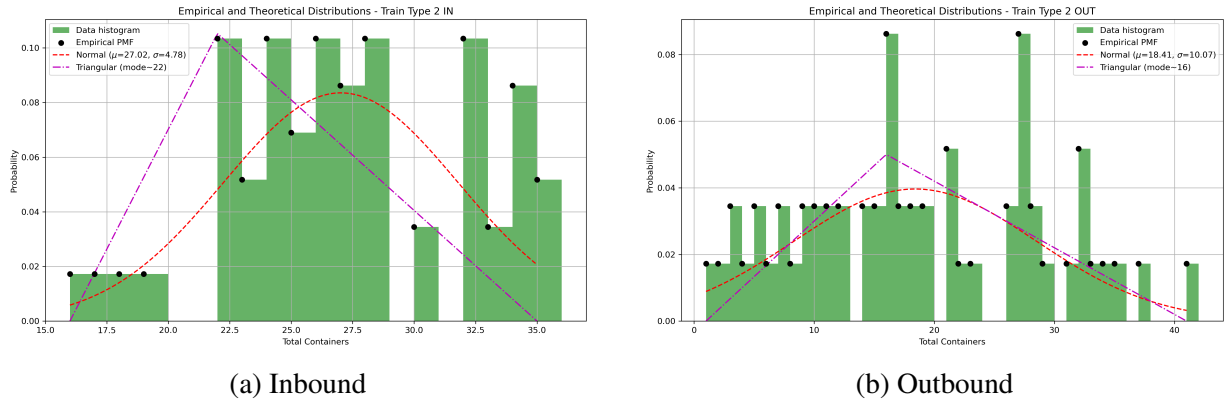


Figure 9.10: Distribution of containers for Train Type 2: (a) inbound trains and (b) outbound trains.

Operational logic of the RMG dispatchers:

The RMG dispatcher is responsible for managing the assignment of handling tasks to the RMG cranes operating in the rail yard. In the simulation model, these tasks correspond to the processing of trains located on the inside tracks. The dispatcher determines which train will be handled next and which RMG crane will perform the operation.

Different task-selection policies are implemented in the model depending on the scenario considered. The simplest policy follows a First-In-First-Out (FIFO) rule, where trains are processed in the order in which they request service. In more advanced scenarios, a priority rule based on the departure time of the trains is used, giving priority to trains that are scheduled to leave earlier.

Another important aspect of the dispatcher logic concerns the number of RMG cranes operating in each park and the management of operational breaks. As described in the previous chapter, operations at PSA GP are organized in 6-hour shifts. Between shifts, two break periods are scheduled: 15 minutes at the end of a shift and 15 minutes at the beginning of the next one. The simulation model explicitly incorporates these shift breaks by temporarily interrupting RMG operations.

This behavior is implemented by adding delay tasks to the RMG operators during the break periods. Two different situations may occur. The first case arises when a train requests an RMG while a shift break is already in progress. In this situation, a delay task is assigned to the RMG operator with a duration equal to the remaining time until the end of the break.

The second case occurs when an RMG crane is actively processing a train when the shift break begins. In this case, a delay task corresponding to the full break duration is added to the RMG operator using a preempt-only rule. This causes the current handling task to be temporarily interrupted until the break period ends.

The first situation is implemented through a user event that is triggered at the beginning of each

shift break. This event sends a message to the RMG operators instructing them to add the corresponding delay task. The second situation is handled directly within the dispatcher logic, specifically in the section of the code where tasks are assigned to the RMG cranes.

Operational logic of the locomotive operators:

Locomotives are modeled as operators responsible for moving trains between the outside tracks and the inside tracks of the rail yard. In particular, locomotives transport trains from the outside tracks to the inside tracks when trains enter the yard, and from the inside tracks to the outside tracks when they leave the terminal.

Similarly to the RMG operators described previously, locomotives are also subject to operational constraints related to shift breaks. The same logic of temporarily interrupting operations or delaying task assignments during break periods is applied.

When a locomotive is already performing a movement and a shift break occurs, the operation is interrupted through a delay task added in the locomotive message trigger. Conversely, when a train requests a locomotive during a shift break, the assignment is postponed by adding a delay task following the same logic implemented in the locomotive dispatcher.

Another aspect of the locomotive logic is implemented through the operator triggers *On Finish* and *On Message*. These triggers are used both to update the labels associated with trains and tracks and to enforce additional operational rules for locomotive movements.

In particular, when a locomotive delivers a train from the outside tracks to the inside tracks, it cannot immediately be assigned to another train. Before serving a new request, the locomotive must return empty to the outside area. This return movement is modeled by introducing a delay that represents the travel time required for the locomotive to reposition itself. During this period, the locomotive remains unavailable for other assignments.

Furthermore, since the locomotive is initially selected when the train moves from the outside track to the intermediate track, additional logic ensures that the same locomotive continues to handle the subsequent movements of that train, namely from the intermediate track to the inside track. A similar mechanism is applied for trains leaving the yard, guaranteeing that the same locomotive performs the sequence of movements required for the outbound operation.

9.2.2 Simulation experiments based on the theoretical model

In this section, the simulation analysis is developed starting from the theoretical model previously presented. In particular, the theoretical train arrival schedule derived from the analytical formulation is used as input to the simulation model. This approach allows the evaluation of how the optimal theoretical schedule, and the corresponding number of trains, are affected when

additional operational constraints, such as locomotive availability, are introduced.

More specifically, the source of the simulation model is initialized using the ideal train arrivals obtained from the optimal solution of the mathematical model presented in Section 9.1.2.2. In particular, by referring to the third row of Table 9.1, it can be observed that, under optimal conditions and assuming always-available locomotives, a maximum of 8 trains can be handled within a 24-hour time horizon.

Based on this optimal solution, the first simulation experiment is designed in FlexTerm by replicating the same arrival schedule. The arrival times and the assigned inside tracks for each train are reported in Table 9.3, where time is expressed in seconds.

It is important to note that all other elements and assumptions of the simulation model remain consistent with those of the theoretical model. In particular, each train requires five hours for handling operations, one hour for entering the terminal, and one hour for departure. Furthermore, the optimization model assumes the availability of a single RMG crane in each park for train handling operations.

Table 9.3: Train schedule in source

Arrival Time [s]	Train Name	Quantity	Inside Track
0	Train1	1	qE_2
0	Train2	1	qW_8
7200	Train3	1	qE_4
10800	Train4	1	qW_7
14400	Train5	1	qE_1
25200	Train6	1	qE_6
28800	Train7	1	qW_1
57600	Train8	1	qW_8

To conduct the experiment, multiple scenarios are analyzed by varying two key parameters of the model.

The parameters considered in the simulation experiments are *AvailabilityRate* and *timeToAvailability*. Meanwhile, Table 9.4 summarizes the key simulation parameters that are kept constant throughout the experimental analysis.

The parameter *AvailabilityRate* represents the probability used in a Bernoulli distribution to determine whether a locomotive is available at the time it is requested. When a train located in an external queue requests a locomotive, the dispatcher evaluates its availability based on the outcome of this probabilistic check. If a locomotive is available and all other operational conditions are satisfied, it is assigned to the train. Otherwise, the train remains in the queue for a duration

Table 9.4: Key simulation parameters and selected configurations (Theoretical experiment)

Parameter	Selected setting (interpretation)
InsideTrackPolicy	2: assign trains to predefined (planned) inside tracks
ProcessingPolicy	1: process trains immediately upon arrival if resources available
DeparturePolicy	1: trains leave immediately after processing
WestTrackPolicy	1: all West track can be used
DelayTrackPolicy	3: assign predefined track after delay
RMG_West	1: one RMG operating in the West park
RMG_East	1: one RMG operating in the East park
ArrivalType	3: theoretical arrival schedule with identical theoretical trains
ArrivalProcessing	1: process trains immediately upon entering internal queue
LocoReturn	2: locomotive return time is not considered
LocoNr	2: number of locomotives
ShiftBreaks	1: not considered

defined by the parameter *timeToAvailability*, which represents the time required for a locomotive to become available again.

Different combinations of these two parameters are used to define the simulation scenarios analyzed in this study.

Table 9.5: Experimental scenarios defined by the combination of *AvailabilityRate* and *timeToAvailability*.

Scenario	AvailabilityRate [%]	timeToAvailability [s]
Scenario 1	100	0
Scenario 2	50	10800
Scenario 3	50	21600
Scenario 4	75	10800
Scenario 5	75	21600

The different experimental configurations are summarized in Table 9.5. Scenario 1 corresponds to the ideal case in which locomotives are always available. Under this assumption, the system is expected to achieve the theoretical capacity identified by the optimization model, processing up to eight trains within a 24-hour time horizon.

The remaining scenarios consider reduced locomotive availability. In particular, *AvailabilityRate*

values of 50% and 75% are combined with *timeToAvailability* values of 10800 and 21600 seconds, corresponding to 3 and 6 hours, respectively. These durations can be interpreted as representing half or one full work shift, based on the 6-hour shift structure adopted at the PSA GP terminal. In this context, a shift denotes a predefined 6-hour operational period during which a specific workforce is assigned.

For each scenario, 20 replications were performed. This replication process accounts for the stochastic nature of locomotive availability, ensuring that the observed performance differences are not driven by random fluctuations. Consequently, it enables statistically robust comparisons across scenarios and provides more reliable estimates of key performance indicators, such as the average number of trains completed within the 24-hour simulation period.

Meanwhile, Fig. 9.11 summarizes the number of trains completed within the 24-hour simulation period across all experimental scenarios. The boxplots represent the distribution of the results obtained from 20 independent replications for each scenario, represented by the dots in the figure. The central line indicates the median, the box edges correspond to the 25th and 75th percentiles, and the whiskers represent the full observed range.

In Scenario 1, where a locomotive is always available, the system consistently handles eight trains per day, confirming the correspondence between the simulation and the theoretical maximum capacity obtained from the optimization model.

As locomotive availability decreases, the achievable capacity declines correspondingly. Under a 50% availability rate (Scenarios 2 and 3), the number of completed trains drops to an average of six and four, respectively, with increased variability as downtime extends from three to six hours. When availability improves to 75% (Scenarios 4 and 5), throughput partially recovers, reaching an average of seven trains when downtime is limited to three hours and decreasing again when locomotives remain unavailable for six hours.

Fig. 9.12 illustrates the Gantt charts of train movements for two representative scenarios. The first case corresponds to the optimal condition, in which the maximum number of eight trains is completed (Fig. 9.12a), while the second case represents a low-performance replication of Scenario 3, where only four trains are completed (Fig. 9.12b).

In the optimal case, all trains are successfully processed within the simulation horizon. The time spent in the outside tracks during entry and exit operations appears negligible, as no explicit processing time is assigned to these stages in the model. Furthermore, the Gantt chart confirms that trains follow their predefined routing, reaching the assigned internal tracks as determined by the input schedule.

In contrast, the low-performance scenario highlights the impact of reduced locomotive availability. A significant amount of time is spent by trains waiting in the outside tracks before entering the yard, due to the lack of available locomotives. Additionally, for the trains that are not completed within the simulation horizon, long waiting times can be observed in the inside tracks. In these cases, although handling operations may be finished, trains are unable to exit the yard due

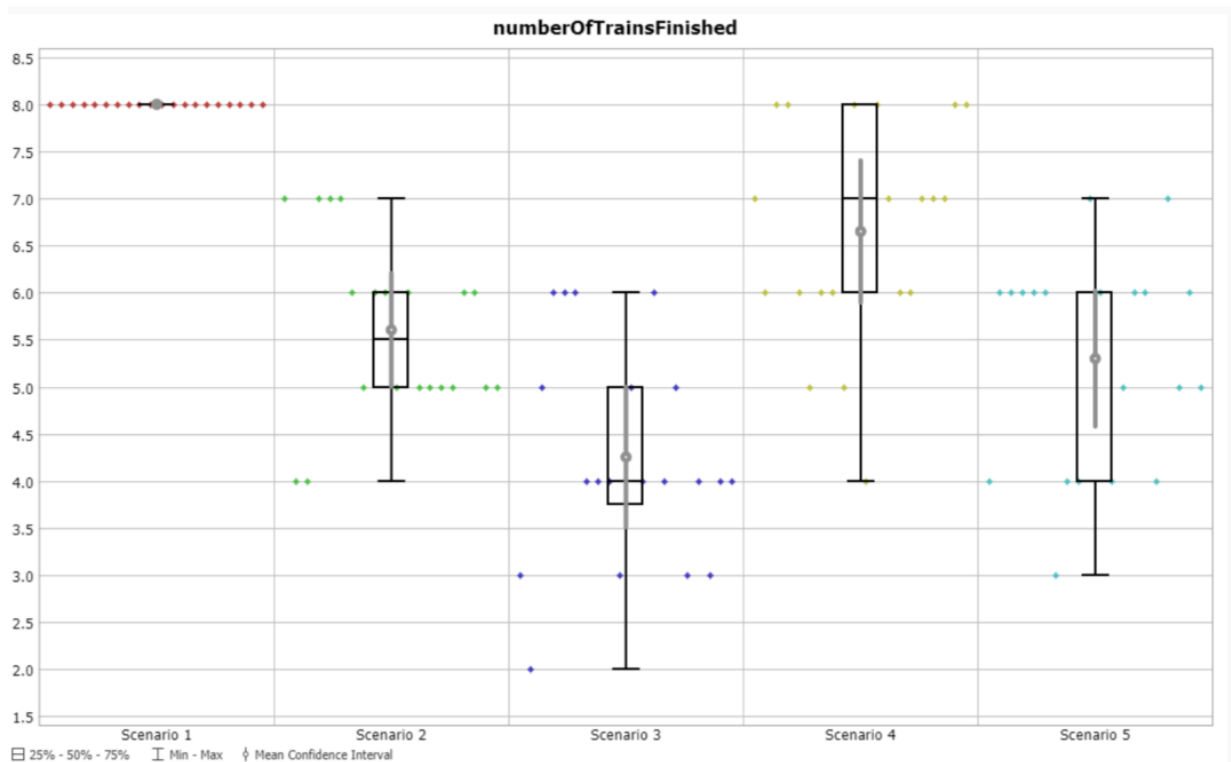
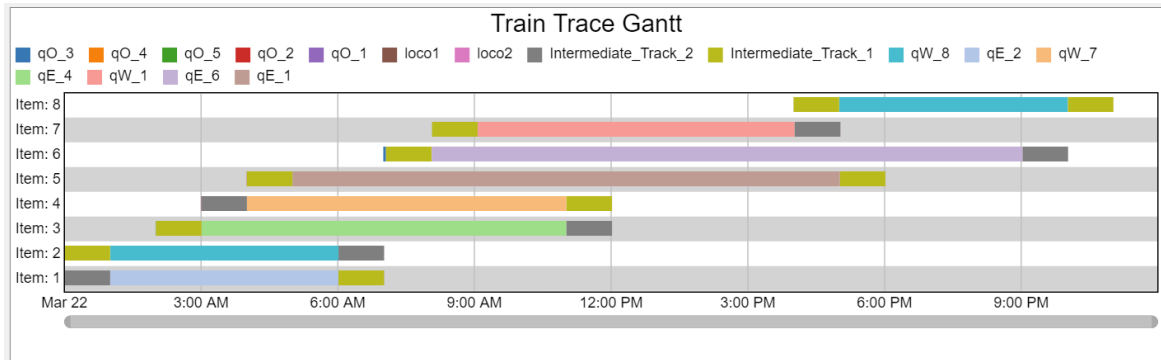


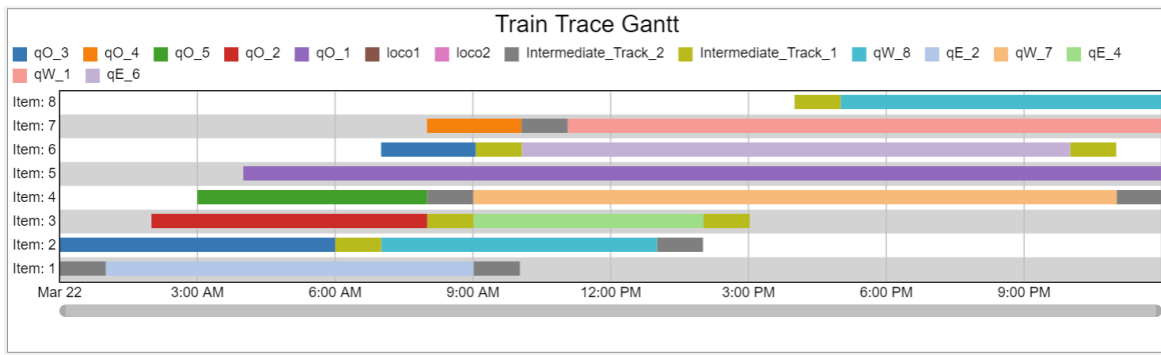
Figure 9.11: Number of trains handled within a 24-hour period under different simulation scenarios (Theoretical experiment)

to the unavailability of locomotives.

The resource utilization statistics further support these observations. In Scenario 1, both RMG cranes are utilized simultaneously for approximately 83.4% of the simulation time, while locomotives are utilized for about 33.7% of the time. In Scenario 3, locomotive utilization decreases, with one locomotive used approximately 30% of the time and the other about 25%. Similarly, the utilization of the RMG cranes is reduced, reaching approximately 66.7% and 62%, respectively.



(a) Representative replication under ideal conditions (Scenario 1)



(b) Low-performance replication (4 trains) under reduced locomotive availability (Scenario 3)

Figure 9.12: Comparison of train movement Gantt charts under different locomotive availability conditions (Theoretical experiment)

9.2.3 Simulation experiments based on the theoretical model with realistic elements added

In order to evaluate the system under more realistic operating conditions, the assumptions of the theoretical model were refined by introducing additional operational details derived from real terminal data.

In particular, several process components that were previously neglected or modeled with fixed durations were replaced by stochastic or data-driven representations. These include train movement times, handling operations, preparation activities, and operator availability constraints.

Table 9.6 summarizes the main differences between the theoretical and realistic configurations of the model.

As shown in Table 9.6, the realistic model introduces additional processing times and stochas-

Table 9.6: Comparison between theoretical and realistic modeling assumptions

Phase	Theoretical model	Realistic model
Outside queue (before entry)	0	Loco Release + loco coupling time before entry
Entrance movement (Intermediate track)	1 h	Triangular distribution based on real data; higher for East-track movements
Before handling (inside track)	0	Maneuvering and loco uncoupling time
Handling (discharge + load)	5 h	Based on number of containers (empirical distribution) and triangular handling time per container
Before exit (inside track)	0	Preparation time (coupling, pressurization)
Exit movement (Intermediate track)	1 h	Triangular distribution based on real data; higher for East-track movements
Outside queue (after exit)	0	Uncoupling, checks, brake test, and final operations
Operators (RMG, locomotives)	Shift breaks not considered	Subject to shift breaks and availability constraints
Locomotive return	Not modeled	Return-empty time explicitly modeled

tic variability across all phases of the train lifecycle. In contrast to the theoretical formulation, where several operations are neglected or represented by fixed durations, the realistic configuration incorporates operational procedures derived from terminal practice, including preparation activities, maneuvering operations, and resource-related constraints.

Moreover, the introduction of stochastic elements in both movement and handling times, together with constraints such as locomotive availability and shift breaks, provides a more accurate representation of real terminal operations. While these factors would typically be expected to reduce the achievable system capacity and increase performance variability, they also introduce a greater degree of flexibility in system dynamics.

An additional difference with respect to the theoretical model concerns the characteristics of the arriving trains. In the realistic configuration, trains are no longer identical but are generated based on real data. In particular, the number of containers to be discharged and loaded is sampled from empirical distributions, and the handling time for each container is drawn from triangular distributions, as described previously.

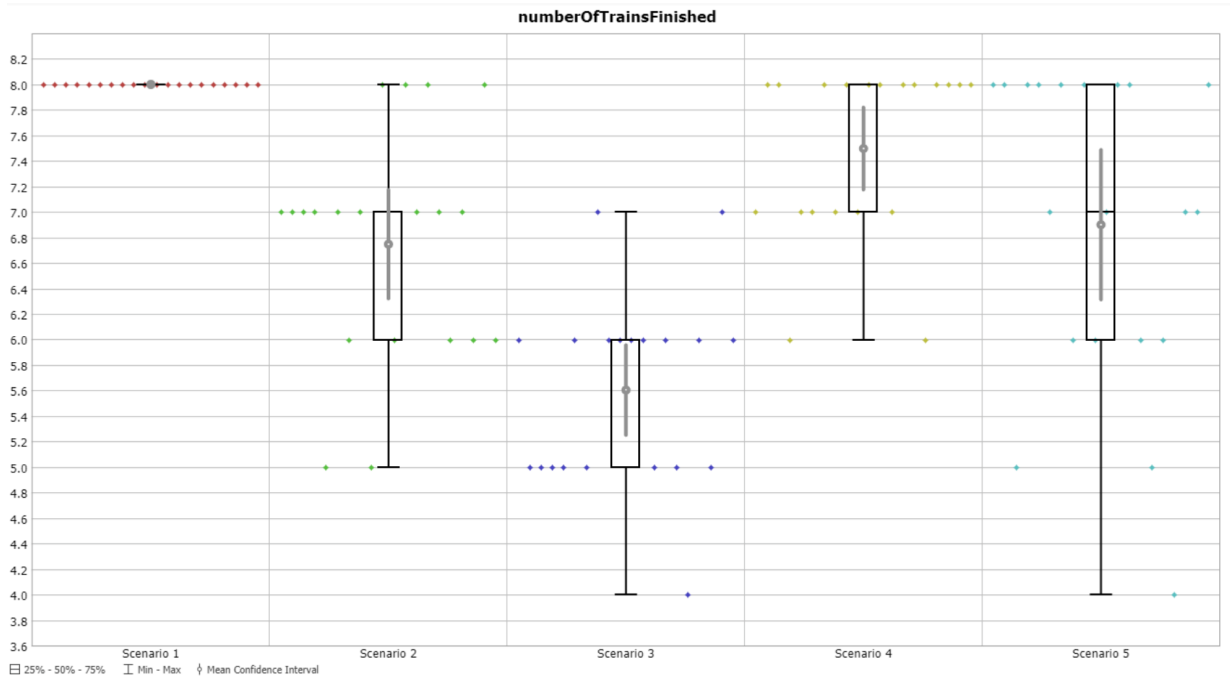


Figure 9.13: Number of trains handled within a 24-hour period under different simulation scenarios (Theoretical experiment with realistic operational parameters)

To ensure a consistent comparison with the theoretical model, the same experimental scenarios are considered (see Table 9.5), and the same arrival schedule and track assignments are used (see Table 9.3). The only difference lies in the introduction of realistic operational elements and heterogeneous train characteristics. The results of these experiments are presented in Fig. 9.13.

For the remaining parameters, which are kept constant throughout the experimental analysis, refer to Table 9.4. The only differences concern the parameters *ArrivalType*, which is set to 3 (corresponding to a theoretical arrival schedule with realistic trains), and *ShiftBreaks*, which is set to 2, indicating that shift breaks are considered in this case.

The results obtained under realistic operating conditions show an improvement in system performance compared to the previous experiment based on fixed processing times. In particular, a higher number of trains is completed on average across most scenarios.

This improvement can be attributed to the introduction of variability in processing and movement times. In the theoretical model, fixed durations lead to synchronized operations and increased competition for shared resources, which may result in congestion. In contrast, the stochastic nature of the realistic model produces a more distributed use of resources over time. In particular, some trains are completed earlier than the average handling time, allowing inside tracks to be freed sooner and enabling earlier requests for locomotives. This reduces blocking effects within

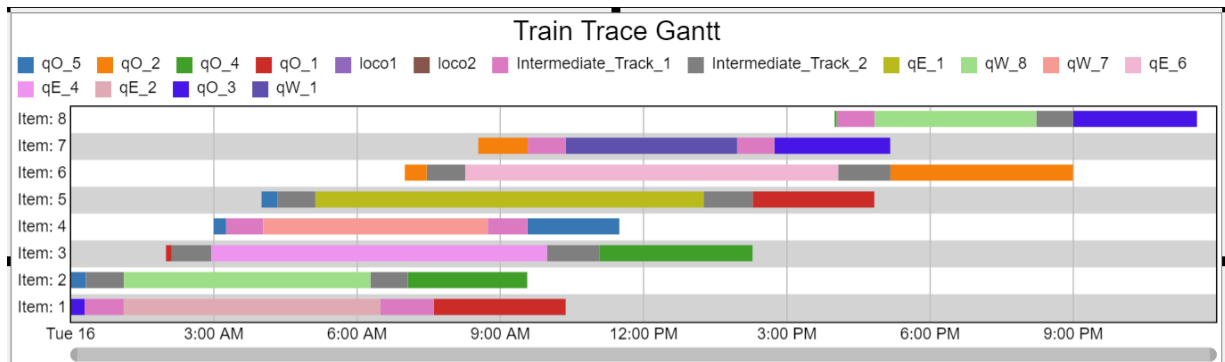


Figure 9.14: Trace Gantt of the trains showing the time trains spent in each of the elements of the model

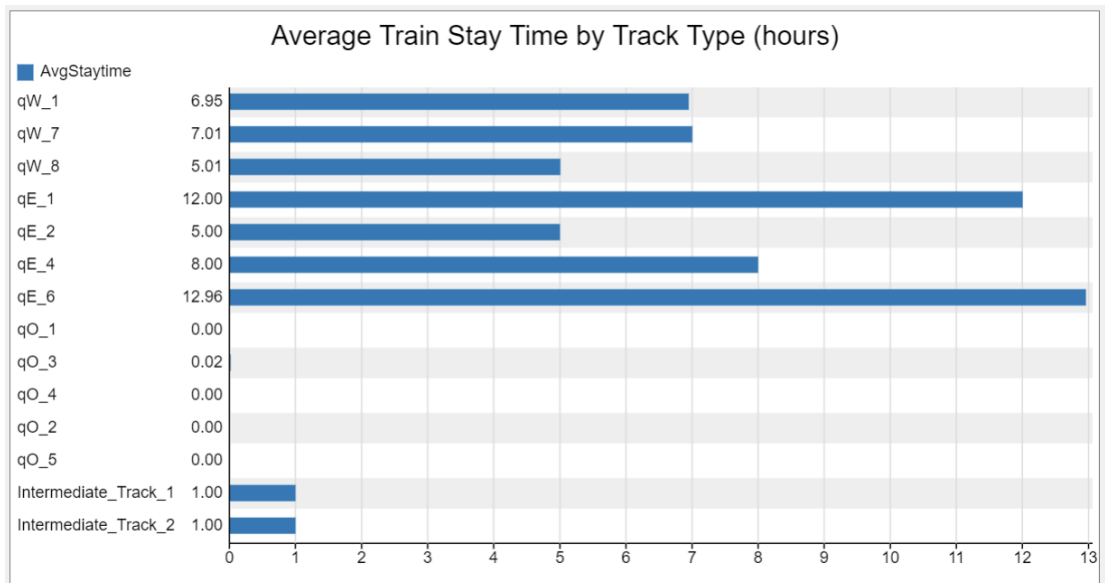
the yard and leads to a more efficient utilization of both locomotives and RMG cranes.

Furthermore, variability reduces the likelihood of multiple trains competing simultaneously for the same resources, thereby smoothing system operations and improving overall throughput. As a result, the system is able to handle a higher number of trains under realistic conditions, despite the presence of additional operational constraints.

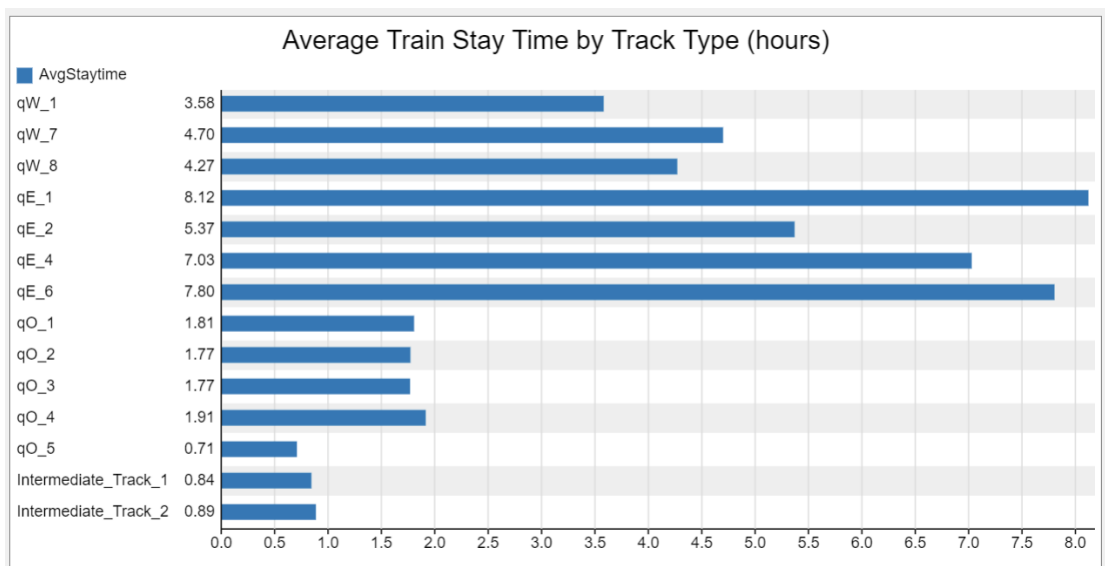
Fig. 9.14 illustrates the distribution of train processing times across the different tracks for one replication of Scenario 1. It can be observed that handling times in the inside tracks are generally shorter and more variable compared to the fixed durations assumed in the theoretical model. At the same time, additional time is now spent in the outside tracks, both during entry and exit, due to operational procedures such as locomotive coupling, decoupling, and verification activities.

Although eight trains are still completed within the 24-hour horizon, a slight difference can be observed: the last train remains in the outside track at the end of the simulation period and has not fully exited the system. This contrasts with the theoretical case, where all trains complete their full cycle within the time horizon.

In terms of resource utilization, locomotive usage remains relatively similar to the previous experiment, with values of approximately 26% and 33% of the simulation time. In contrast, RMG utilization decreases to approximately 57% and 54%, reflecting the reduced and more variable handling times associated with the realistic modeling of train operations.



(a) Average train stay time by track types (theoretical model)



(b) Average train stay time by track type (model with realistic elements added)

Figure 9.15: Average stay time of trains across different track types under theoretical and realistic modeling assumptions

Fig. 9.15 compares the average stay time of trains across the different track types for the theoretical model and the the one where realistic elements are added.

In the theoretical configuration, the majority of the time is spent in the inside tracks, particularly

in the East park, where significantly higher values are observed. This is mainly due to the use of fixed processing times and because of the delay caused when a locomotive is busy. In contrast, the model with realistic elements added exhibits a more balanced distribution of time across the different track types. While the stay time in the inside tracks is reduced, additional time is now observed in the outside tracks, both during entry and exit. This is due to the inclusion of operational activities such as locomotive coupling, decoupling, and verification procedures.

Overall, these results highlight how the introduction of realistic operational elements redistributes processing time across the system, reducing congestion in the inside tracks and contributing to a smoother and more efficient flow of trains through the terminal.

Although the theoretical model provides an upper bound on system capacity under idealized assumptions, it is based on deterministic and highly structured operations. This rigidity may lead to synchronized resource usage and increased congestion, which can negatively affect system performance.

In contrast, the introduction of variability in the simulation model allows for a more flexible and realistic representation of operations. This flexibility reduces synchronization effects and improves the utilization of resources, resulting in higher observed throughput in some scenarios.

9.2.4 Simulation experiments under model saturation

The objective of this set of experiments is to evaluate the maximum operational capacity of the rail yard under saturated demand conditions. To this end, a large number of train arrivals is introduced into the system, with all trains entering the source at time instant 0. This configuration ensures that the system remains continuously loaded, allowing the assessment of how many trains can be processed within a given time horizon under different configurations.

As input to the model, 200 train arrivals are considered, corresponding to 200 trains entering the terminal. These arrivals are generated based on real operational data collected over a one-week period. Since the number of trains observed in a single week is not sufficient to saturate the system, the same weekly arrival pattern is replicated until a total of 200 trains is obtained.

The purpose of these experiments is to evaluate the impact of both resource capacity and operational constraints on the overall system performance.

A baseline configuration is first defined. In this configuration, trains are allowed to be processed immediately upon arrival, without waiting for the beginning of the next operational shift. Similarly, trains are allowed to leave the system as soon as processing is completed and the required resources are available, without being constrained by a predefined departure schedule. The system is configured with two locomotives, and both shift breaks and locomotive return times are considered. One track in the West park, specifically the uppermost track, is always kept free in order to ensure connectivity between the West and East parks; this track is therefore used exclu-

sively for train movements and not for handling operations. The assignment of trains to inside tracks is performed randomly. This selection policy is also applied when a train is delayed at the source and must request access to the system again after a waiting period.

Starting from this baseline configuration, the experiments are structured to evaluate the impact of different parameters. In particular, the first set of experiments focuses on analyzing the effect of varying the number of RMG cranes on the system capacity.

9.2.4.1 Changing the number of the RMGs

The experiment investigates the effect of varying the number of RMG cranes on the number of trains handled, while all other parameters are kept fixed as reported in Table 9.7.

Table 9.7: Key simulation parameters and selected configurations for the under-saturation model with different numbers of RMGs

Parameter	Selected setting (interpretation)
InsideTrackPolicy	1: assign trains in a random inside tracks
ProcessingPolicy	1: process trains immediately upon arrival if resources available
DeparturePolicy	1: trains leave immediately after processing
WestTrackPolicy	2: leave one West track free
DelayTrackPolicy	1: assign trains in a random inside tracks after delay
ArrivalType	1: arrival schedule for model saturation
ArrivalProcessing	1: process trains immediately upon entering internal queue
LocoNr	2: number of locomotives
ShiftBreaks	2: considered
Loco AvailabilityRate	100: always available

Meanwhile the scenarios based on the parameters changed are presented in the Table 9.8.

Table 9.8: Experimental scenarios for RMG configuration and locomotive return policy

Scenario	RMG_West	RMG_East	LocoReturn
Scenario 1	1	1	1 (Yes)
Scenario 2	1	2	1 (Yes)
Scenario 3	2	1	1 (Yes)
Scenario 4	2	2	1 (Yes)
Scenario 5	1	1	2 (No)
Scenario 6	1	2	2 (No)
Scenario 7	2	1	2 (No)
Scenario 8	2	2	2 (No)

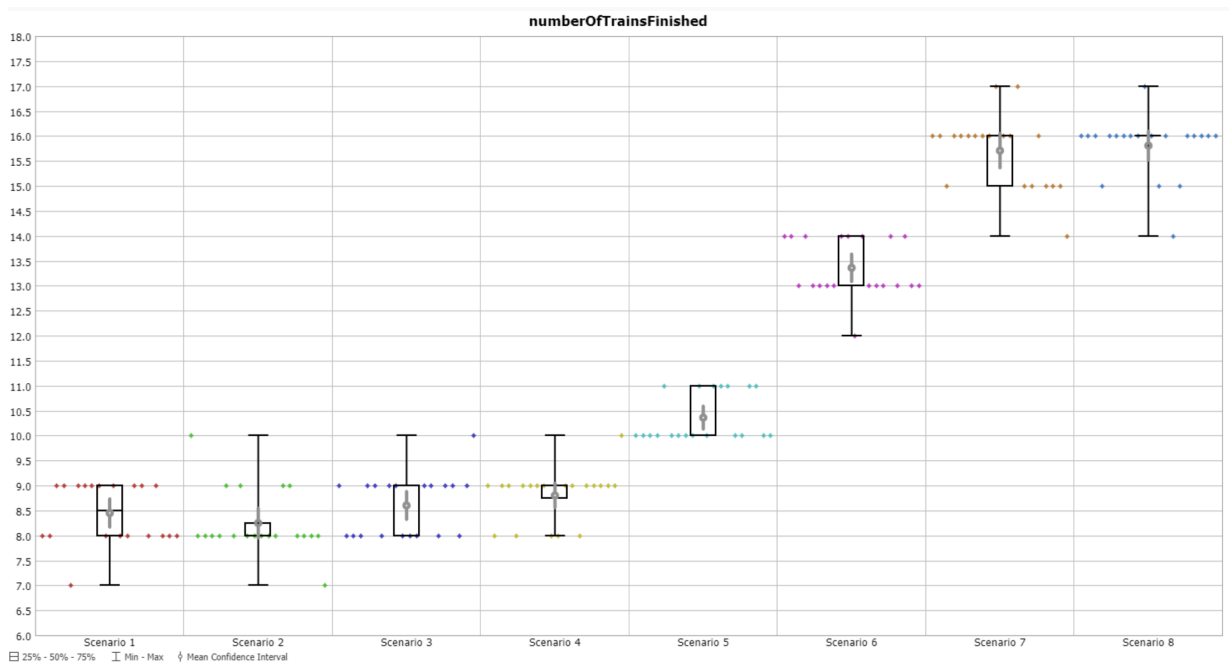


Figure 9.16: Number of trains handled over a 24-hour period under different RMG configurations

Eight scenarios are defined based on the combination of two factors: the number of RMG cranes operating in each park and whether the locomotive return time is considered.

The results are presented in Fig. 9.16. In the first four scenarios, where the locomotive return time is considered, the impact of increasing the number of RMGs is not particularly significant. In these cases, the number of trains handled remains relatively low, with a maximum of approximately 9 trains and only occasional instances reaching 10.

This limited performance, even when two RMG cranes are available, can be attributed to the constraints imposed by the locomotive return logic. In particular, once a locomotive completes a

task, it must spend additional time returning before it can be assigned to a new train. During this period, the locomotive remains unavailable, reducing the system's effective transport capacity. Furthermore, locomotive movements occupy critical infrastructure such as intermediate tracks and the connecting West track, temporarily preventing their use by other trains and increasing congestion within the system.

In contrast, in the last four scenarios, where the locomotive return time is not considered, the impact of the number of RMG cranes becomes much more evident. In this case, the system achieves significantly higher throughput, with the maximum number of completed trains reaching up to 16 when two RMG cranes are available in each park. When only one RMG crane per park is used, the number of completed trains ranges approximately between 10 and 11. Therefore, under saturated conditions and neglecting locomotive return time, this configuration represents the effective upper bound of system capacity.

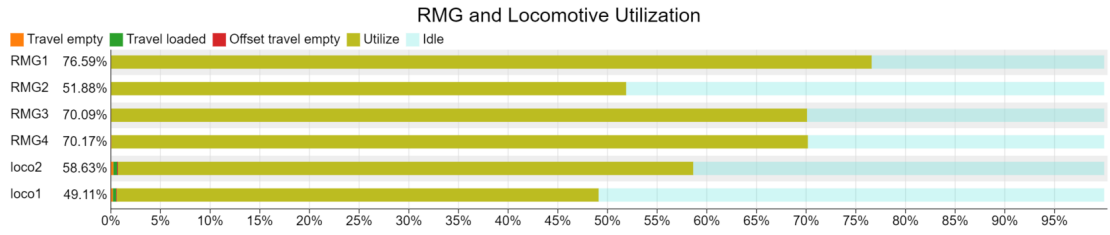
A difference can also be observed between Scenarios 6 and 7. In Scenario 7, where two RMG cranes operate in the West park, a slightly higher number of trains is completed compared to Scenario 6, where two RMG cranes are instead located in the East park. This behavior is explained by the system layout: trains handled in the East park must pass through a West track to exit the terminal. As a result, even if trains are fully processed in the East park, they may not be counted as completed if a West track is not available for their departure.

It should be noted that this effect is particularly visible because the simulation horizon is limited to one day. Over longer time horizons, where more trains are able to complete their full cycle and exit the system, the difference between these scenarios becomes negligible, typically affecting only the last train in the system.

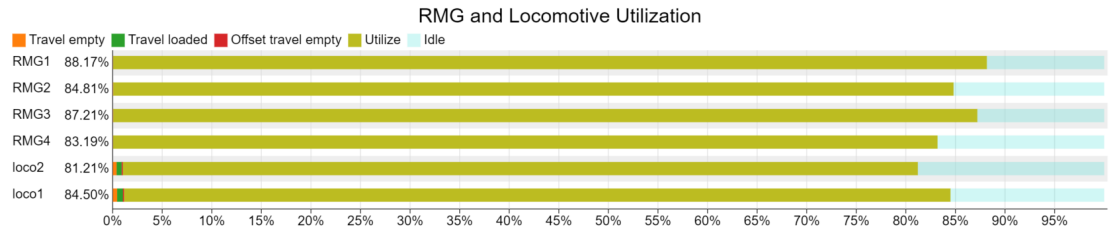
To further investigate the impact of locomotive availability on the overall system capacity, the utilization of both RMG cranes and locomotives is analyzed for two representative scenarios: Scenario 4, where locomotive return time is considered, and Scenario 8, where this constraint is removed. These scenarios are selected as they represent extreme configurations in which two RMG cranes are available in each park. The analysis focuses on specific replications whose behavior is consistent with the majority of the observed outcomes, corresponding to cases where 9 and 16 trains are completed, respectively.

The results are presented in Fig. 9.17.

In Scenario 4, locomotive utilization is relatively limited, with values around 50-60%, while RMG utilization ranges between approximately 50% and 76%. This indicates that the system is not able to fully exploit its handling capacity. The reason lies in the locomotive return logic: after completing a transport task, locomotives require additional time to return before they can serve another train. As a result, trains experience delays both in entering and exiting the system, reducing the frequency at which RMGs receive new work. Consequently, RMG cranes remain partially idle despite the availability of handling capacity.



(a) Scenario 4 – With locomotive return time



(b) Scenario 8 – Without locomotive return time

Figure 9.17: Resource utilization under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time

In contrast, in Scenario 8, where locomotive return time is not considered, locomotive utilization increases significantly, reaching values of around 80-85%. This leads to a more continuous flow of trains within the system. As a result, RMG utilization also increases, reaching values above 80%, indicating a much more efficient use of handling resources.

However, it is important to note that the increase in RMG utilization is not proportional to the increase in the number of completed trains. While throughput nearly doubles, RMG utilization increases only moderately. This is because, even when handling operations are completed, trains may still remain in the system due to residual constraints related to locomotive movements and track availability.

The temporal evolution of resource activity is illustrated in Fig. 9.18. The shift breaks, occurring at 05:45–06:15, 11:45–12:15, 17:45–18:15, and 23:45–00:15, can be clearly identified as periods during which both RMG cranes and locomotives are inactive.

These interruptions are particularly evident in Scenario 8, where the resources operate continuously outside the shift breaks. In contrast, in Scenario 4, additional idle periods can be observed, especially for the RMG cranes after completing certain trains, due to the lack of available trains to be processed.

Similarly, locomotives in Scenario 4 exhibit extended idle times as a consequence of the return logic: after completing a movement, they are temporarily unavailable for new tasks, and this state is recorded as idle rather than as empty travel. This further contributes to the reduced system throughput observed in this configuration.

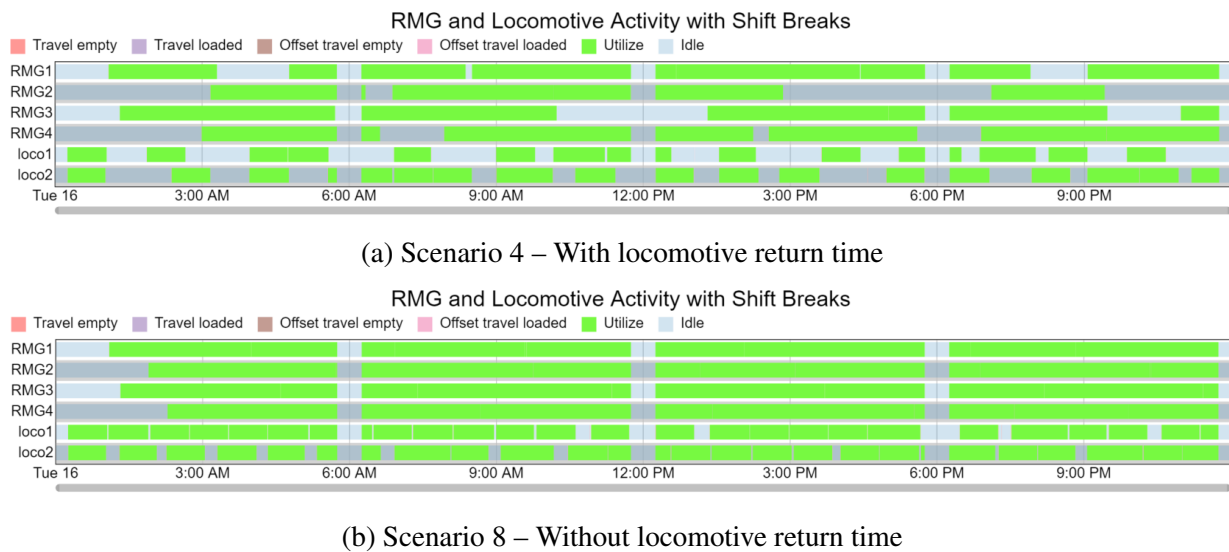


Figure 9.18: Resource activity over time under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time

Overall, these results highlight that the system bottleneck is primarily associated with transport operations rather than handling capacity. When locomotive return constraints are present, the system is transport-limited, whereas removing this constraint allows the system to better exploit its handling resources, shifting the bottleneck towards RMG capacity.

In order to obtain a more robust evaluation of system capacity, the same set of experiments is repeated over a longer time horizon, specifically one week of operations. This allows a more accurate assessment of the impact of resource configurations, as the system processes a significantly larger number of trains.

In the one-day experiments (Fig. 9.16), the differences between configurations may appear limited. For instance, the effect of using one or two RMG cranes in the East park is not always clearly visible. However, this is mainly due to the short time horizon, where a small number of trains (e.g., one or two) may be completed but not yet exited the system, thus affecting the measured throughput.

Over a longer time horizon, these transient effects become less significant. As the total number of handled trains increases, the results provide a more representative picture of system performance. In this case, the impact of resource configurations becomes clearer and can be interpreted more reliably.

The results of the one-week simulation are presented in Fig. 9.19. Compared to the one-day experiments, the impact of the different configurations becomes clearer and more consistent.

In particular, the results of Scenario 7 are now much closer to those of Scenario 6, as expected,

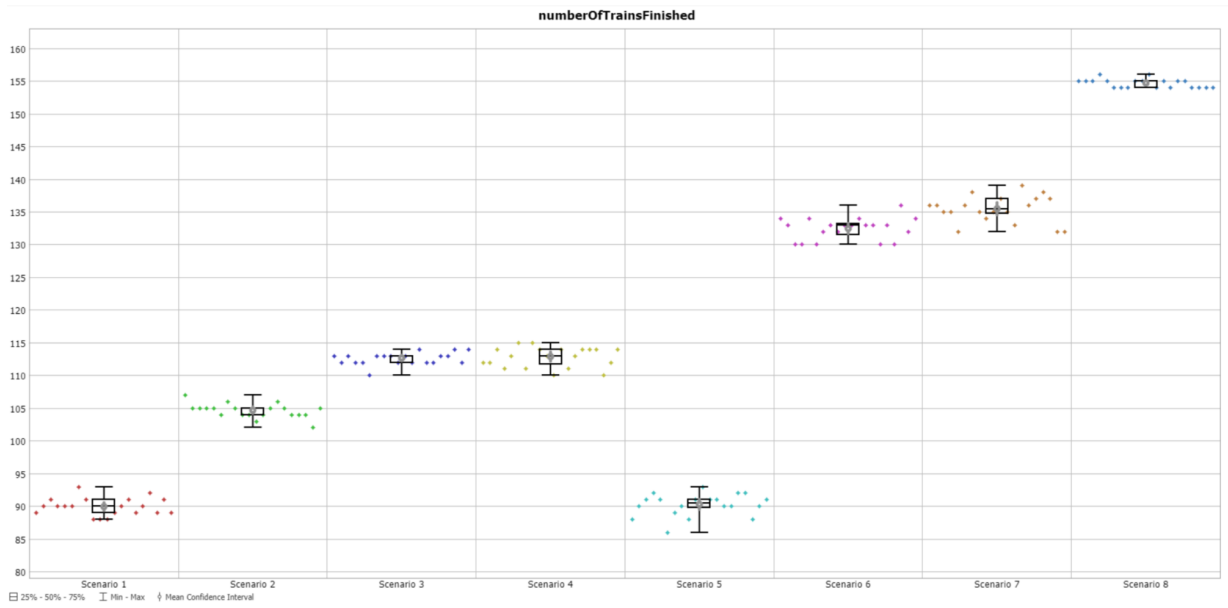


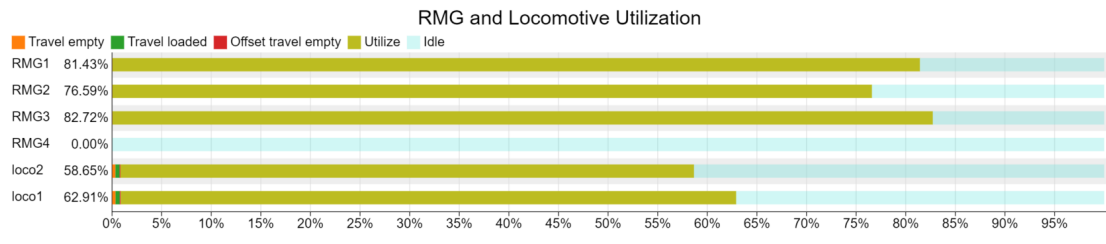
Figure 9.19: Number of trains handled over one week under different RMG configurations

rather than being closer to Scenario 8. This confirms that, over a longer time horizon, transient effects observed in shorter simulations are reduced, allowing for a more reliable comparison of system performance.

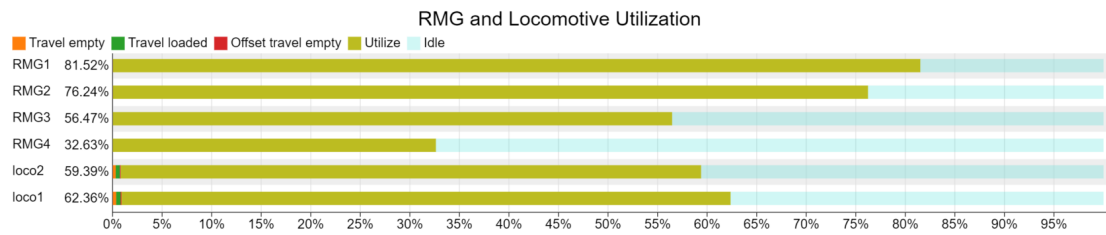
It can also be observed that a slightly better performance is achieved when two RMG cranes are deployed in the West park rather than in the East park. This is explained by the system layout: trains processed in the East park must pass through a West track to exit the terminal. Therefore, even when trains are fully handled in the East park, their departure may be delayed due to the additional constraint of requiring a free West track, preventing them from being counted as completed.

On the other hand, when the locomotive return time is considered, Scenario 4 (with two RMG cranes in each park) does not show a significant improvement compared to Scenario 3 (with two RMGs in the West park and one in the East). This further confirms that locomotive availability plays a more critical role than handling capacity. In such cases, the system is transport-limited, and increasing the number of RMG cranes does not lead to a proportional increase in throughput.

This behavior is further supported by the utilization results shown in Fig. 9.20. In Scenario 4, the additional RMG introduced in the East park remains almost unused, while the utilization levels of the other cranes are comparable to those observed in Scenario 3. This indicates that the system is not able to continuously supply trains to all available cranes. In particular, the limited availability of locomotives restricts the flow of trains entering and leaving the inside tracks, preventing the effective exploitation of the additional handling capacity.



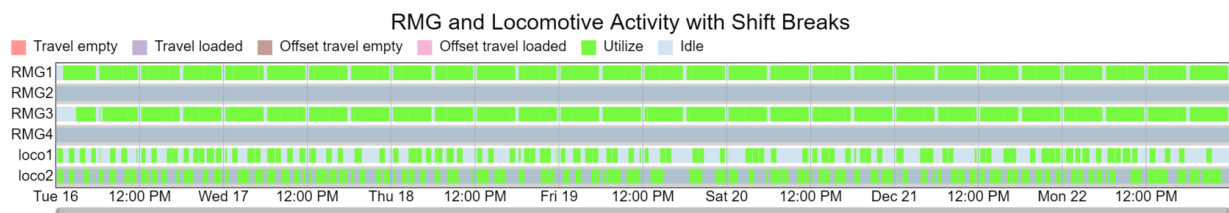
(a) Scenario 3 – 2 RMGs in West park and 1 in the East one



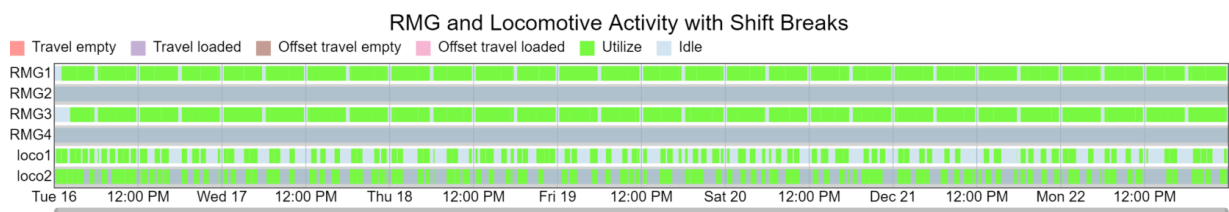
(b) Scenario 4 – 2 RMGs in each park

Figure 9.20: Resource utilization under saturated conditions for two representative scenarios with varying numbers of RMG cranes

Finally, Scenarios 1 and 5, where only one RMG crane is available in each park, exhibit very similar performance, with an average of approximately 90 trains processed per week, even when locomotive return time is not considered. This behavior can be explained by analyzing the utilization of RMG cranes, which operate at nearly full capacity (around 90%, as shown in Fig. 9.22b) in both scenarios. In addition, Fig. 9.21 confirms that RMG cranes are continuously active throughout the simulation period in both configurations. As a result, the system becomes handling-limited: since RMGs are already fully utilized, locomotive availability does not significantly affect overall performance.

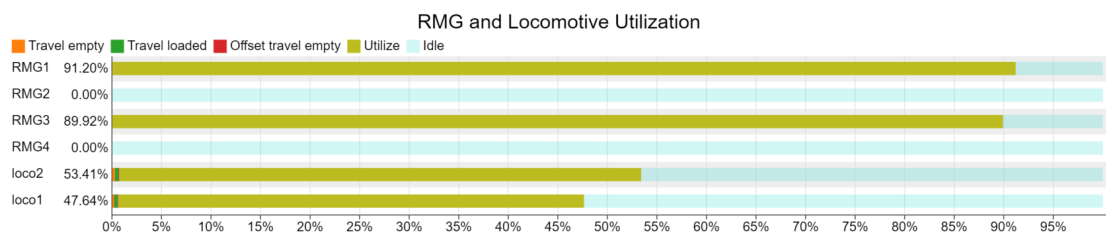


(a) Scenario 1 – With locomotive return time

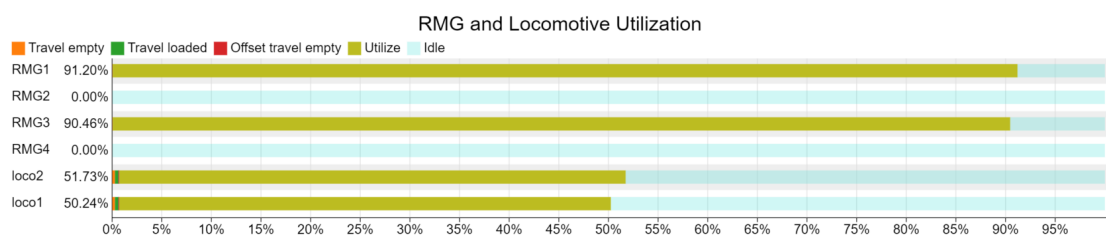


(b) Scenario 5 – Without locomotive return time

Figure 9.21: Resource activity over time under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time over a one-week simulation horizon



(a) Scenario 1 – With locomotive return time



(b) Scenario 5 – Without locomotive return time

Figure 9.22: Resource utilization under saturated conditions for two representative scenarios considering the presence or absence of locomotive return time over a one-week simulation horizon

For the remaining scenarios, the observed trends can be directly explained by the previously identified system mechanisms, and additional graphical analysis is therefore omitted for clarity.

Table 9.9: Number of completed trains over one week for different scenarios

Scenario	Mean Trains	Std. Dev.
1	90	1.5
2	105	1.6
3	112	1.5
4	113	1.6
5	90	2.0
6	132	2.0
7	136	2.2
8	155	1.2

To complement the graphical analysis, Table 9.9 summarizes the average number of completed trains for each scenario over the one-week simulation horizon. The results confirm the trends observed in Fig. 9.19, providing a clearer quantitative comparison between configurations. It can be observed that the highest performance is achieved in Scenario 8, where both handling and transport constraints are minimized, while Scenarios 1 and 5 show similar performance due to the system being handling-limited. Intermediate scenarios highlight the impact of locomotive constraints and system layout on overall throughput.

These results highlight the combined effect of resource availability and system constraints on terminal capacity. In the following subsection, the analysis is extended by investigating the impact of operational constraints, while keeping the number of resources fixed.

9.2.4.2 Changing operational constraints

In this part, the experiments aim to evaluate the impact of operational constraints on the overall system capacity, in contrast to the previous analysis, where the focus was on the effect of resource configurations.

The parameters that remain fixed throughout these experiments are summarized in Table 9.10. The different scenarios considered are presented in Table 9.11, and are defined based on combinations of the parameters *Processing Policy* and *Loco Return*. The *Processing Policy* determines whether a train can be processed immediately upon arrival, provided that resources (RMG cranes) are available, or whether it must wait until the next shift. The *Loco Return* parameter specifies whether the time required for a locomotive to return after completing a task is considered or neglected.

Table 9.10: Key simulation parameters and selected configurations for the under-saturation model under varying operational constraints

Parameter	Selected setting (interpretation)
InsideTrackPolicy	1: assign trains in a random inside tracks
DeparturePolicy	1: trains leave immediately after processing
WestTrackPolicy	2: leave one West track free
DelayTrackPolicy	1: assign trains in a random inside tracks after delay
ArrivalType	1: arrival schedule for model saturation
ArrivalProcessing	1: process trains immediately upon entering internal queue
LocoNr	2: number of locomotives
RMGNr West Park	1: number of RMGs
RMGNr East Park	1: number of RMGS
ShiftBreaks	2: considered
Loco AvailabilityRate	100: always available

Table 9.11: Operational policy scenarios under saturated conditions

Scenario	Processing Policy	Loco Return
1	1 (Immediate)	1 (Yes)
2	1 (Immediate)	2 (NO)
3	2 (Next shift)	1 (Yes)
4	2 (Next shift)	2 (No)

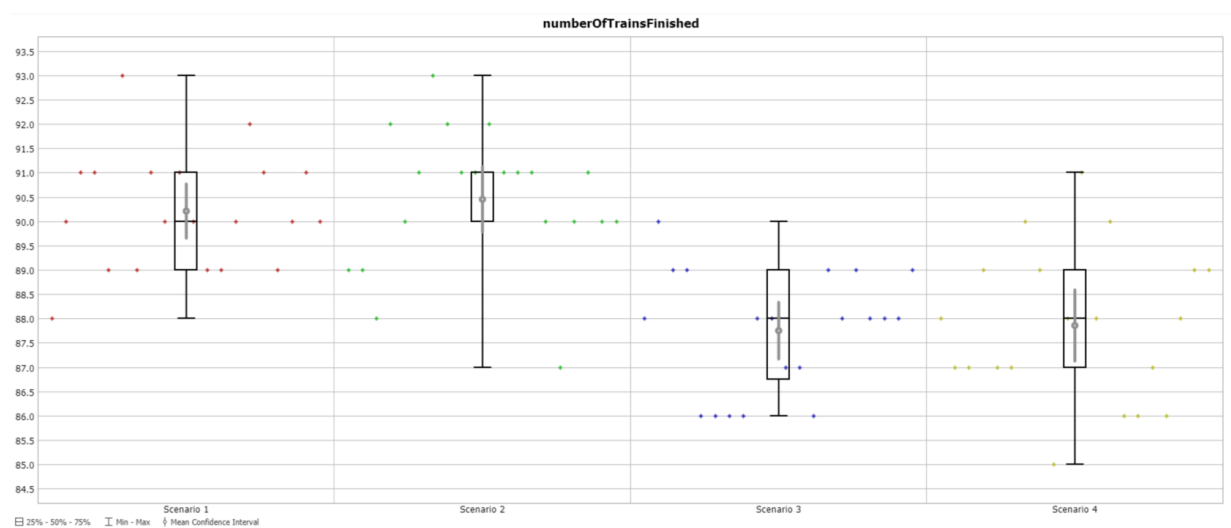


Figure 9.23: Number of trains handled over one week under different operational policies

As shown in Fig. 9.23, the requirement that trains can only be processed starting from the next shift (considering a shift duration of 6 hours) has its impact on system performance. This can be observed by comparing Scenario 1 and Scenario 3, where, on average, approximately 2-3 additional trains are completed per week when immediate processing is allowed.

This effect is mainly due to the delay introduced at the beginning of the simulation, when trains must wait until the next shift before being processed. Over a one-week horizon, this delay has a limited but still visible impact, as it primarily affects the initial system response and, to a lesser extent, the completion of the last trains within the simulation period.

On the other hand, the impact of the locomotive return time is less significant. This can be observed by comparing Scenario 1 with Scenario 2, and Scenario 3 with Scenario 4, where only minor differences in the number of completed trains are observed. This indicates that, under the considered configuration, the locomotive return constraint has a reduced influence on overall system capacity compared to the processing policy.

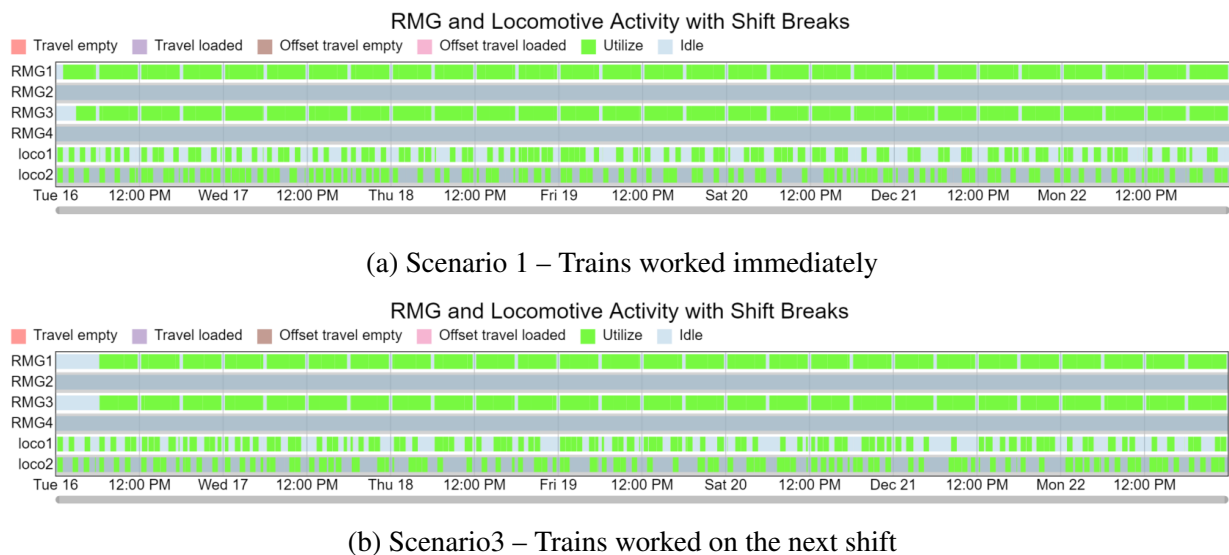


Figure 9.24: Resource activity over time under saturated conditions for two scenarios with different processing policies

Fig. 9.24 illustrates the resource activity over time for two representative scenarios. This figure highlights the impact of the processing policy on the initial utilization of RMG cranes.

In Scenario 3, where trains can only be processed starting from the next shift, RMG cranes remain idle during the first shift despite the presence of trains in the system. In contrast, in Scenario 1, RMGs begin operating immediately as soon as trains enter the terminal and are ready to be processed.

This initial delay explains the difference in overall performance between the two scenarios. In

terms of utilization, RMG cranes in Scenario 1 exhibit slightly higher values (approximately 2–3%) compared to Scenario 3. This difference arises because, over the same simulation horizon, RMGs in Scenario 1 are able to start processing earlier, leading to a more efficient use of available time.

Another experiment was conducted to evaluate the effect of relaxing the constraint that trains can only be processed starting from the next shift. In particular, a *Gray Interval* of one hour was introduced. This parameter allows trains arriving within the first hour of a shift to be processed immediately, without waiting for the next shift. All other parameters remain as defined in Table 9.10, with the processing policy set to 2 (i.e., processing starts in the next shift) and the locomotive return time considered.

Meanwhile Table 9.12 summarizes the scenarios considered to evaluate the effect of introducing a Gray Interval.

Table 9.12: Scenarios for the evaluation of processing flexibility through a Gray Interval

Scenario	Processing Flexibility
1	1 (Strict next-shift processing)
2	2 (Next-shift with 1-hour Gray Interval)

The results, shown in Fig. 9.25, indicate a slight improvement in the number of completed trains when the constraint is relaxed. However, the effect is limited due to the relatively short duration of the Gray Interval and the small number of trains arriving within this time window. A larger Gray Interval would likely lead to a more significant increase in system performance. This result highlights that even small relaxations of operational constraints can improve system performance, although the magnitude of the effect depends on how frequently the condition is triggered.

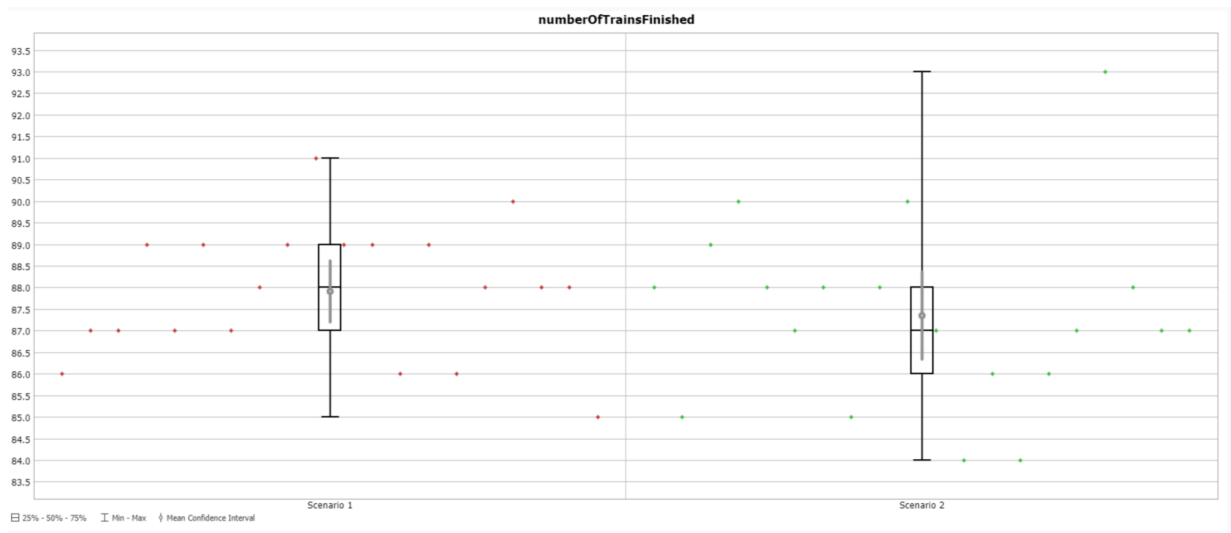


Figure 9.25: Number of trains handled over one week under different operational policies under a relaxed processing policy

Related publications. Parts of the work presented in this chapter have been published in [125] and [126].

Chapter 10

Conclusions and Future Directions

This chapter concludes the thesis by summarizing the main findings of the research and drawing overall conclusions. These conclusions are intended to address the research questions formulated in the introduction. In addition, the chapter outlines recommendations and directions for future research, which constitute an important component of the concluding discussion.

As a reminder, the research presented in this thesis focuses on two key inland-side subsystems of the port terminal: the gate and the rail yard. Accordingly, the main findings of the study are primarily related to these two components and their contribution to overall terminal performance.

First and foremost, adopting a system-of-systems perspective proved fundamental in structuring the research and interpreting the results. This perspective enabled a comprehensive understanding of how different terminal subsystems interact and influence one another, while maintaining a specific focus on gate and rail yard operations. As a result, the findings highlight not only the performance of these subsystems in isolation, but also their broader impact on terminal efficiency.

Starting from the prediction models for truck arrivals in the export case, it can be observed that vessel arrival times, and particularly delays in vessel arrivals, directly affect the time window during which the yard is open to receive export containers. Changes in this time window consequently influence the pattern of truck arrivals at the terminal. A similar relationship can be observed for containers leaving the terminal by truck. Therefore, incorporating data related to vessel schedules and delays can contribute to more accurate predictions of container flows entering and leaving the terminal by truck.

The methodology, based on a feedforward neural network, provides satisfactory results. In the export case, the model successfully forecasts the arrival curve of containers in more than 80% of the cases, achieving an R index greater than 0.7. The use of the Leave-One-Out Cross-Validation (LOOCV) technique also proves useful in identifying cases where the model performs less effectively.

For the import case, similar performance is observed: for the largest dataset considered, approximately 70% of the cases exhibit an R coefficient higher than the average and a lower-than-average Mean Squared Error (MSE).

Nevertheless, it should be noted that the proposed methodology is subject to certain limitations. The prediction framework developed in this thesis focuses primarily on forecasting the total number of containers handled per day. The corresponding hourly distribution required for the optimization scheme is subsequently derived using a recurrent curve. While this approach proved effective for daily aggregate predictions, it is less suitable for accurately capturing container volumes at an hourly level.

In particular, the application of an Artificial Neural Network (ANN) for predicting smaller hourly container volumes did not yield the same level of performance as observed for daily totals. A more accurate hourly prediction would likely require the adoption of a different methodological approach, such as time-series-based forecasting techniques.

Furthermore, additional limitations arise when more detailed container-level information is required. Future applications may necessitate predictions not only of total daily volumes, but also of container characteristics, such as the number of full and empty containers or the distribution between 20-foot and 40-foot units. Preliminary tests indicate that the feed-forward ANN approach employed in this study is not sufficiently effective for this level of detailed classification. Addressing these aspects therefore represents an important direction for future research.

Considering the volume of truck traffic at the terminal, on average 2,300 trucks per day, and given that a terminal area can be considered congested when more than eight trucks are simultaneously present, it is evident that uncontrolled truck arrivals can easily lead to congestion. This is particularly critical for container pickup operations in import areas, which constitute the focus of this thesis in relation to congestion optimization.

The optimization scheme proposed in this work aims to mitigate congestion by redistributing only a subset of truck arrivals that are expected to contribute most significantly to congestion. In this context, the implementation of a full Truck Appointment System (TAS) may be challenging, as it requires broad acceptance and coordination among multiple stakeholders, which is often difficult to achieve in practice. As an alternative, a partial appointment or redistribution strategy, applied only to a limited number of containers or trucks, may represent a more feasible and pragmatic solution.

The discussion on the practical implementation of the proposed approach naturally leads to the question of how truck arrivals should be temporally redistributed when congestion is predicted for a given day or time window. This decision must be made while carefully accounting for operational feasibility and yard constraints.

Truck arrivals may be redistributed to earlier time windows, provided that such adjustments fall within the truck's free interval, a condition that is ensured by construction within the optimization

problem and guarantees that the corresponding containers are already available in the yard. Suitable candidates for temporal advancement may therefore include trucks assigned to containers with longer dwell times or to containers that are more conveniently positioned within the stack and require limited rehandling operations. Nevertheless, advancing truck arrivals may introduce additional operational risks, as the associated resources and processes may not yet be fully ready to support an earlier pickup. For this reason, rescheduling trucks to later time windows should generally be preferred whenever feasible, particularly for trucks associated with containers that are operationally demanding to retrieve, such as those located in deeper stack positions or in yard areas with high planned workload.

Overall, these considerations highlight the asymmetric nature of temporal flexibility in truck rescheduling decisions and underscore the importance of embedding operationally grounded decision rules when translating the proposed approach into practice. By focusing congestion mitigation efforts on selected truck arrivals, the proposed approach can be implemented proactively while preserving operational feasibility and service reliability. Nevertheless, despite the potential benefits of such a redistribution strategy, its practical adoption requires non-negligible effort from the terminal, particularly in terms of coordinating and adjusting truck arrival times. This emphasizes the need to strike a careful balance between operational effectiveness and practical implementability when deploying the proposed solution.

A similar redistribution logic to the one proposed for congestion prevention is also applied when assessing the environmental impact of the approach from a sustainability perspective. The emission estimation framework developed in this thesis relies on emission factors provided by the European Environment Agency (EEA), which are derived from the COPERT model. This ensures consistency with the European reference methodology for road transport emission inventories while allowing the model to be applied within the proposed decision-support framework.

The novelty of the proposed approach does not lie in the emission calculation methodology itself, but rather in the use of predicted truck arrivals, more precisely, predicted container flows handled by trucks, as an input for emission estimation. This predictive capability enables emissions to be assessed in advance and makes it possible to evaluate redistribution strategies aimed at avoiding concentration peaks on specific days or time windows. In the present work, the environmental assessment focuses on a specific pollutant, selected based on data availability; however, the proposed framework is readily extendable to additional pollutants provided that the corresponding emission factors are available.

While such redistribution strategies do not necessarily reduce total emissions over the considered planning horizon, since the total number of trucks remains unchanged, they play an important role in mitigating short-term emission peaks. These peaks are typically associated with a temporal concentration of pollutant release, increased idling, and less efficient vehicle operating conditions. By smoothing the temporal distribution of truck arrivals, the proposed approach contributes to a more balanced emission profile over time, which is particularly relevant from a local air quality and environmental sustainability perspective.

Two redistribution methods were proposed for this purpose. The first method was algorithm-based and offers greater flexibility, making it easier to incorporate additional conditions or operational constraints in future extensions. The second method relied on a mathematical optimization formulation. Although the introduction of further constraints in this formulation may increase computational complexity or risk nonlinearity, it provides a more structured and accurate redistribution of truck arrivals for the problem setting considered in this thesis.

For the scope and assumptions addressed in this work, the optimization-based approach emerges as the most suitable solution among the two, offering a cleaner and more precise redistribution while preserving computational tractability. At the same time, the algorithmic approach remains a valuable alternative for scenarios requiring greater adaptability or incremental integration of additional operational considerations.

Moving on to the second subsystem examined in this thesis, particular attention was devoted to rail operations. The performance of the rail park can significantly affect other terminal processes and overall operational efficiency. Improved rail performance, both in terms of optimization and processing speed, can lead to a more effective utilization of terminal handling equipment, thereby enhancing coordination among different terminal operations. In particular, optimizing the scheduling of container discharge from trains can contribute to reducing unnecessary equipment movements within the yard. For example, a more structured discharge sequence can minimize the distance traveled by handling equipment such as Reach Stackers (RS), leading to time savings and, consequently, to lower fuel consumption. These effects translate into both operational and environmental benefits, as reduced travel distances improve productivity while decreasing energy use.

To address the problem of optimizing train discharge operations, two main methodological approaches were developed and analyzed in this thesis. The first approach is based on an exact modeling framework, in which the train discharge process is formulated as a flexible flow shop scheduling problem and subsequently expressed as a mathematical optimization model.

The exact solution of this model led to a reduction of approximately 25% in the total time required to discharge the train, as demonstrated by the results. However, the main limitation of this approach lies in the computational time required to solve larger instances, particularly in the case of trains with a higher number of containers.

For this reason, a heuristic approach based on Tabu Search was developed. This method proved effective in optimizing the scheduling of train discharge operations, for example in the case of a train with 24 containers. Although the time savings achieved with the heuristic approach are relatively lower compared to the exact method, the results show that similar performance can be obtained with reduced resource usage, particularly with a lower number of Prime Movers (PMs).

Nevertheless, both approaches present certain limitations. In particular, simplifications related to distances and vehicle velocities are adopted in the models. Although these assumptions are close to real operating conditions, they may affect the accuracy of the obtained solutions.

To address these limitations, a detailed simulation model was developed and is described in this thesis. The purpose of this model is to evaluate the effectiveness of the proposed optimization solutions under more realistic conditions, accounting for more accurate representations of travel distances and vehicle speeds. The simulation results are not yet available and are therefore left for future work.

This integrated approach makes it possible to address the research question concerning the role of such analyses in supporting decision-making for train discharge operations. In particular, the results of the optimization models highlight the potential of decision-support tools capable of automatically generating improved discharge schedules, thereby facilitating the work of rail planners.

Another important part of this thesis concerns the evaluation of the capacity of the rail yard in terms of the number of trains that can be handled within a given time horizon. To address this problem, two main methodological approaches were proposed.

In the first approach, the problem is formulated mathematically based on queuing theory. The system of train arrivals at the terminal is analyzed by considering physical, operational, and resource constraints, and it is modeled by representing each track as a queue of trains. Due to the complexity of accurately representing all elements of the real system, certain simplifications are introduced. As a result, the estimated capacity represents a theoretical evaluation, which nevertheless provides useful insights into the overall capacity of the yard and the impact of varying key parameters. However, this approach also requires considerable computational effort when multiple scenarios are analyzed.

For this reason, a simulation-based approach is also adopted. Using the FlexTerm environment, a detailed model of the rail yard is developed, allowing the inclusion of more realistic and operationally relevant information compared to the theoretical model. An extensive experimental campaign is conducted, including different types of analyses. First, the performance of the theoretical model is evaluated within the simulation environment under additional constraints. Subsequently, more comprehensive experiments are carried out by incorporating realistic elements and progressively saturating the system in order to assess its maximum capacity. Several scenarios are considered, including variations in the number of resources, changes in operational constraints related to locomotive movements, and different policies governing how and when trains are processed.

The results confirm the importance of key resources, such as Rail Mounted Gantry cranes (RMGs) and locomotives, in determining the total number of trains handled. In addition, they highlight the impact of operational decisions, particularly regarding the timing of train handling, on the overall system performance.

The simulation model proves to be a valuable tool, as it facilitates the inclusion of detailed system characteristics and enables the execution of multiple experiments within relatively short computational times. For instance, simulation runs consisting of 20 replications over a one-week

horizon can be completed within a few minutes. Overall, the simulation framework supports the identification of system bottlenecks and provides a flexible environment to evaluate how changes in both resource availability and operational strategies influence the performance of the rail yard.

Finally, this thesis contributes through the development of prediction, optimization, and simulation models, providing a comprehensive framework for the analysis and improvement of terminal operations. The results demonstrate the potential of these tools to enhance operational efficiency and, most importantly, to support more informed and effective decision-making processes within the terminal.

Several limitations of the present work open directions for future research. From a predictive perspective, one possible extension concerns the development of more detailed models for containers leaving the terminal by truck. In particular, moving from daily to hourly predictions would allow a more accurate representation of truck flows and would improve the effectiveness of redistribution strategies aimed at reducing congestion.

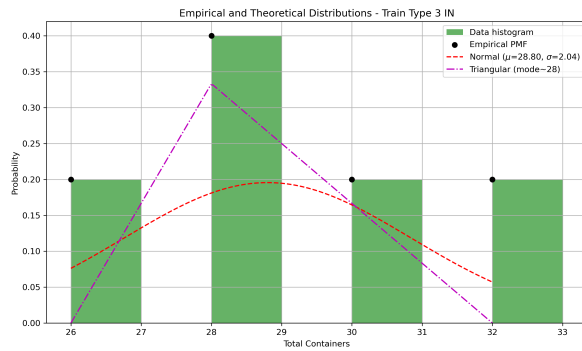
Another important aspect concerns the analysis of the results obtained from the detailed simulation model for train discharge operations. In particular, future work should focus on evaluating the performance of the optimized scheduling solutions within the simulation environment, in order to assess their effectiveness under more realistic operating conditions.

A key direction for future research concerns the practical implementation of the proposed methodologies within terminal operations. Integrating these models into real-world decision-making processes would allow their validation in operational settings and provide insights into their actual impact on terminal performance. This would require additional efforts, particularly in terms of integration with existing information systems and the development of user-friendly interfaces to support terminal operators.

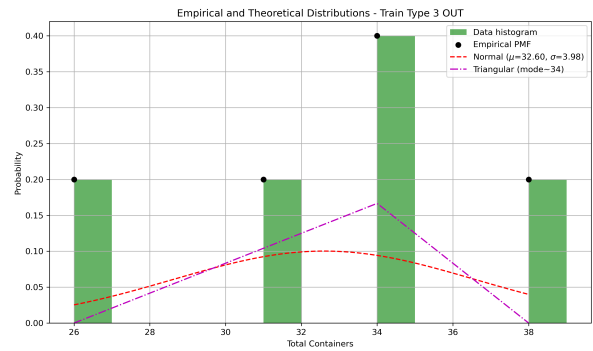
Finally, it is important to note that the models and methods developed in this industrial thesis provide a solid foundation for further extensions. They can be adapted, refined, and expanded to incorporate additional operational requirements and to address new challenges arising in port terminal operations.

Appendix A

The figures below present the distribution analyses for all the other train types considered in Section 9.2.1.3.

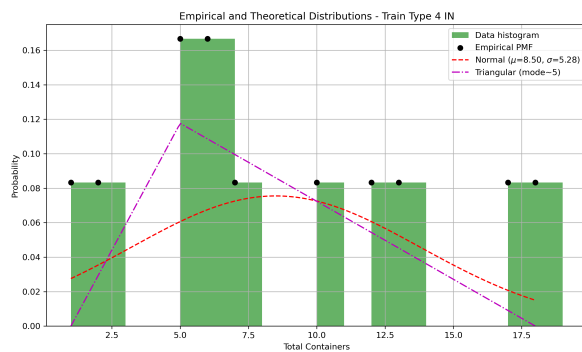


(a) Inbound

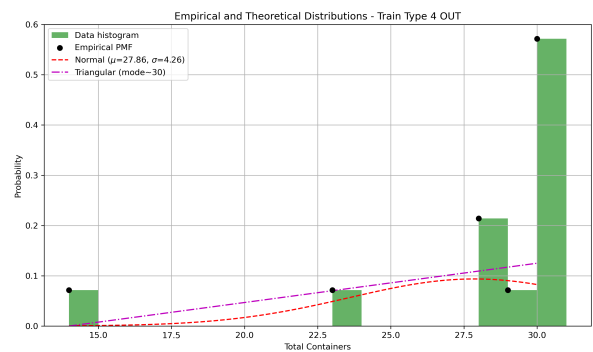


(b) Outbound

Figure 10.1: Distribution of containers for Train Type 3: (a) inbound trains and (b) outbound trains.

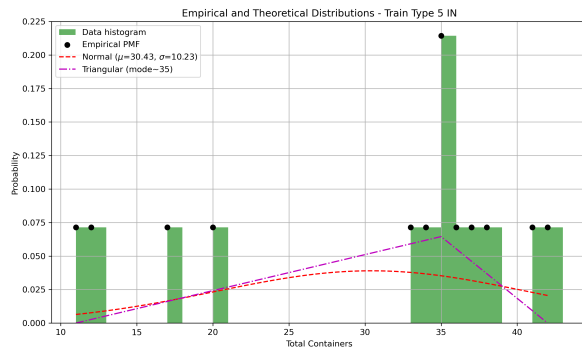


(a) Inbound

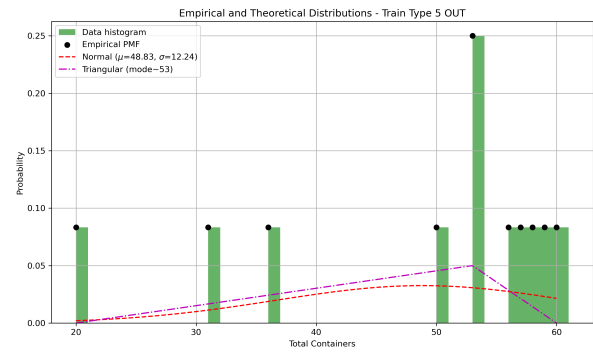


(b) Outbound

Figure 10.2: Distribution of containers for Train Type 4: (a) inbound trains and (b) outbound trains.

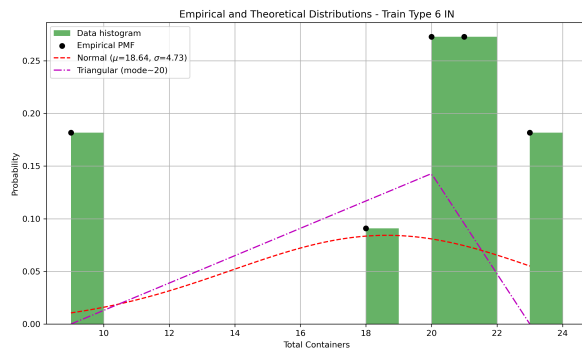


(a) Inbound

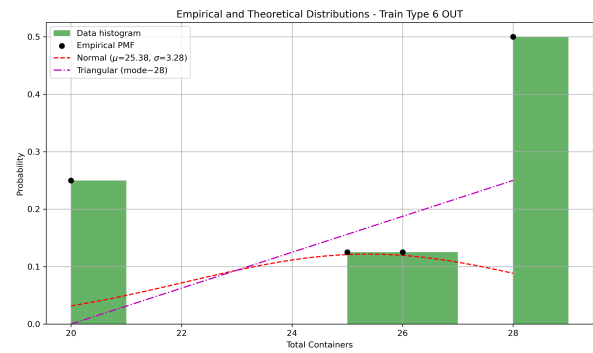


(b) Outbound

Figure 10.3: Distribution of containers for Train Type 5: (a) inbound trains and (b) outbound trains.

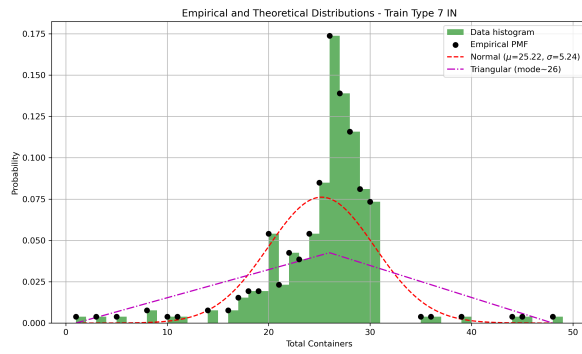


(a) Inbound

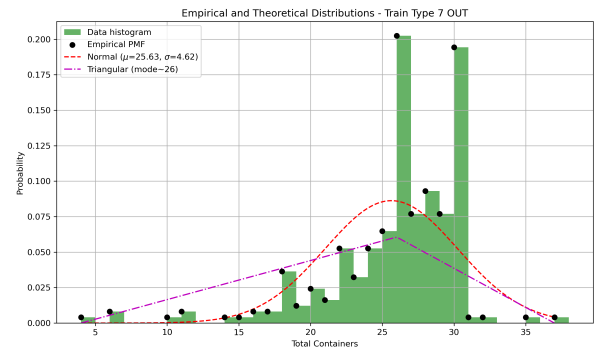


(b) Outbound

Figure 10.4: Distribution of containers for Train Type 6: (a) inbound trains and (b) outbound trains.

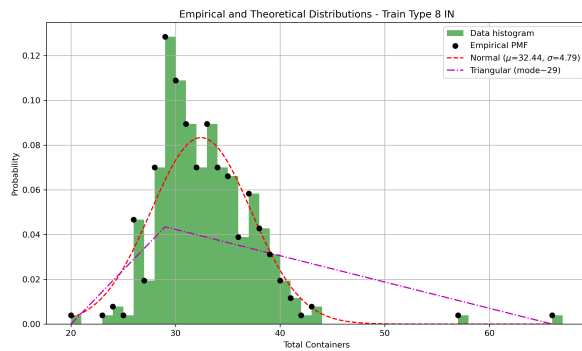


(a) Inbound

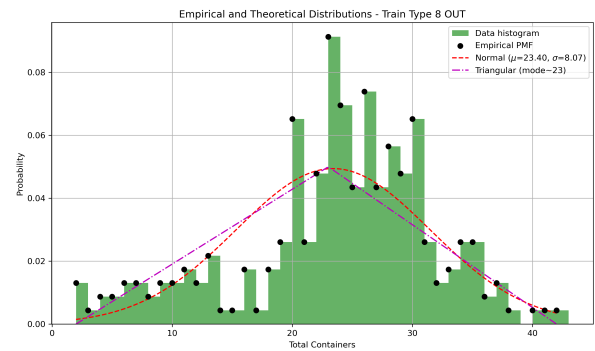


(b) Outbound

Figure 10.5: Distribution of containers for Train Type 7: (a) inbound trains and (b) outbound trains.

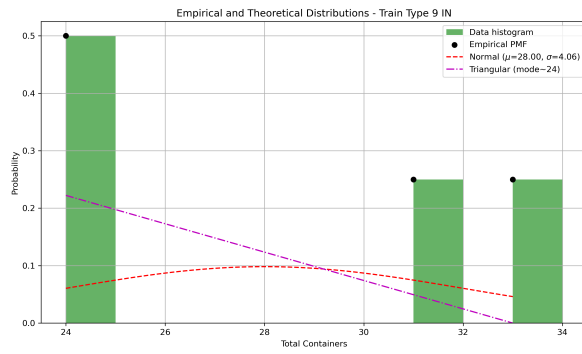


(a) Inbound

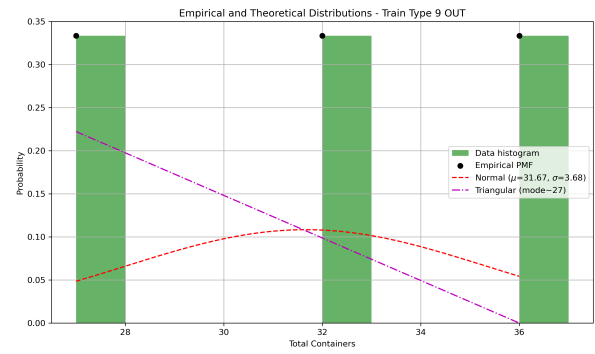


(b) Outbound

Figure 10.6: Distribution of containers for Train Type 8: (a) inbound trains and (b) outbound trains.

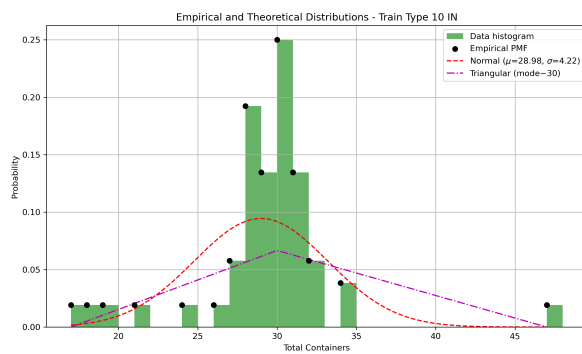


(a) Inbound

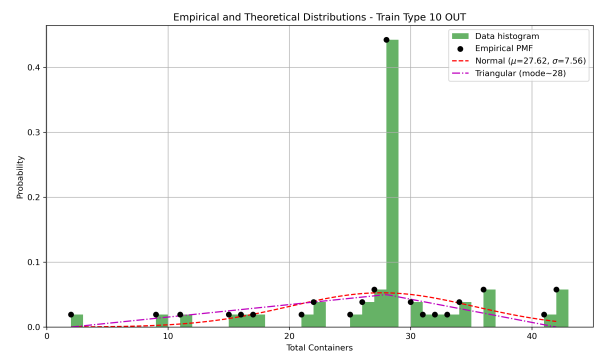


(b) Outbound

Figure 10.7: Distribution of containers for Train Type 9: (a) inbound trains and (b) outbound trains.

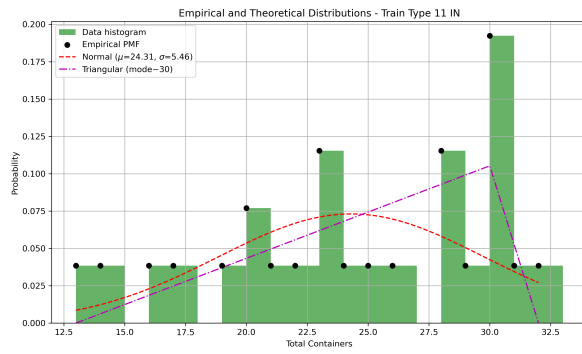


(a) Inbound

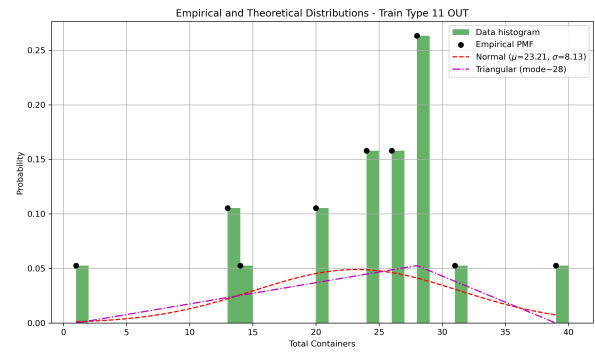


(b) Outbound

Figure 10.8: Distribution of containers for Train Type 10: (a) inbound trains and (b) outbound trains.

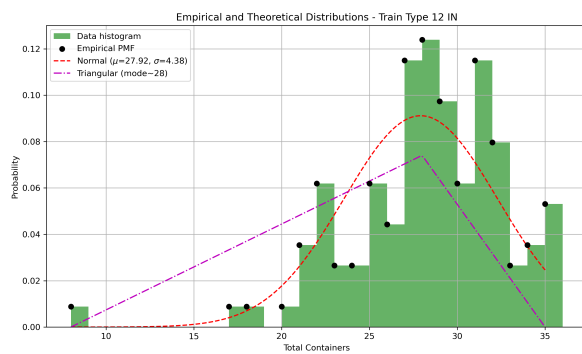


(a) Inbound

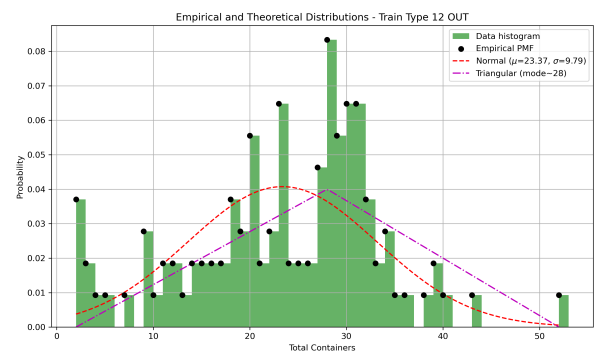


(b) Outbound

Figure 10.9: Distribution of containers for Train Type 11: (a) inbound trains and (b) outbound trains.

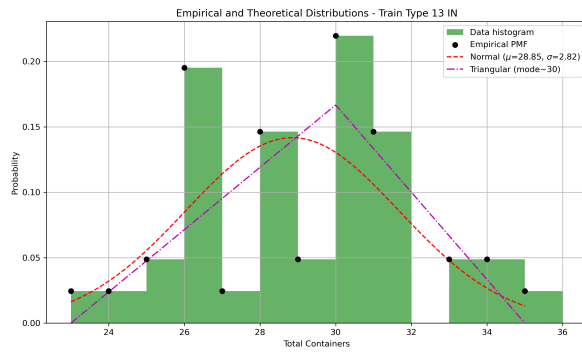


(a) Inbound

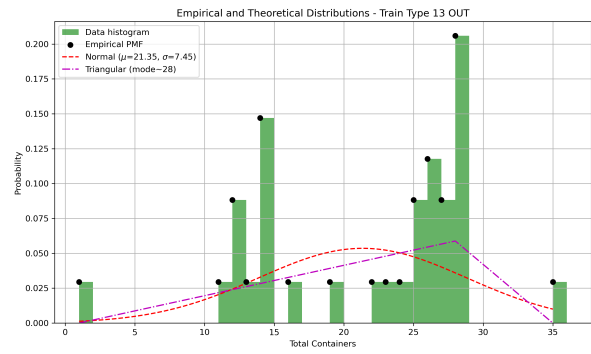


(b) Outbound

Figure 10.10: Distribution of containers for Train Type 12: (a) inbound trains and (b) outbound trains.

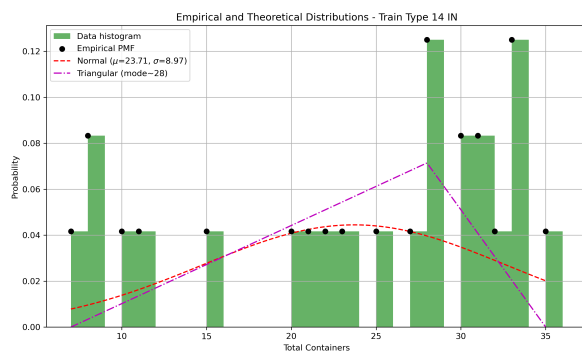


(a) Inbound

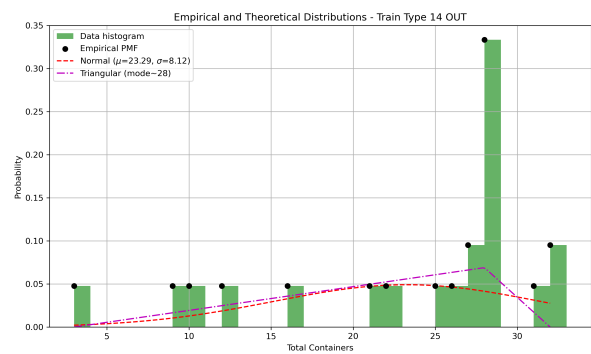


(b) Outbound

Figure 10.11: Distribution of containers for Train Type 13: (a) inbound trains and (b) outbound trains.

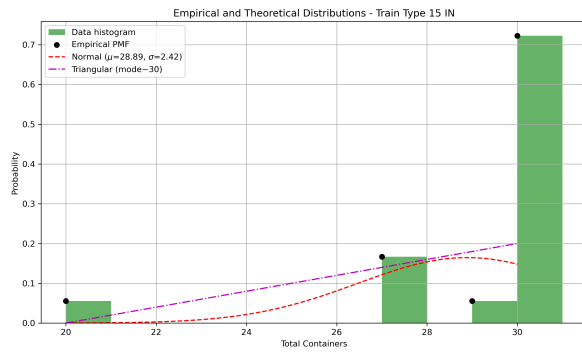


(a) Inbound

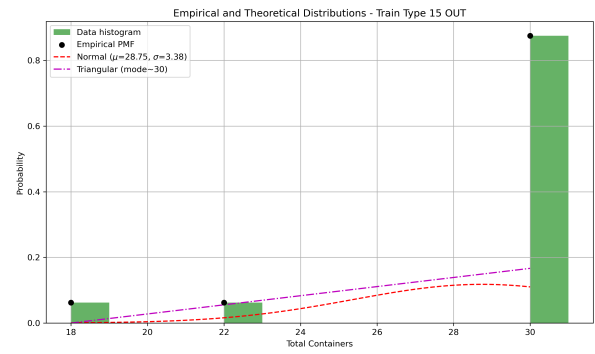


(b) Outbound

Figure 10.12: Distribution of containers for Train Type 14: (a) inbound trains and (b) outbound trains.

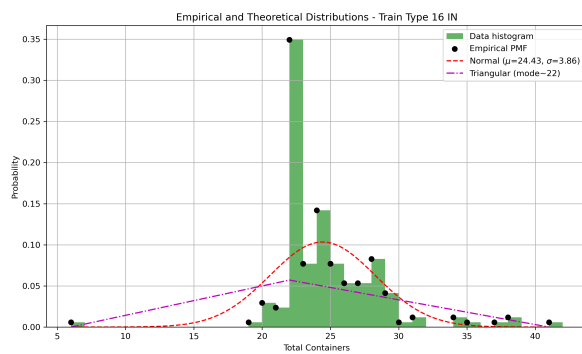


(a) Inbound

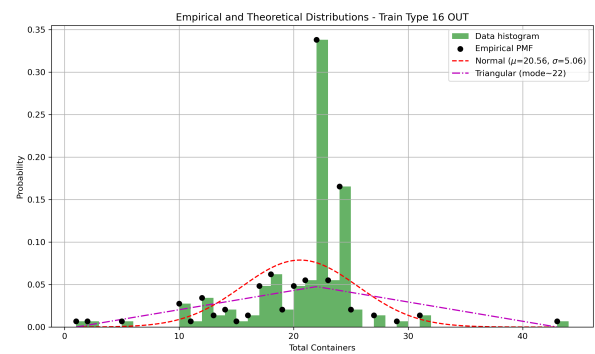


(b) Outbound

Figure 10.13: Distribution of containers for Train Type 15: (a) inbound trains and (b) outbound trains.

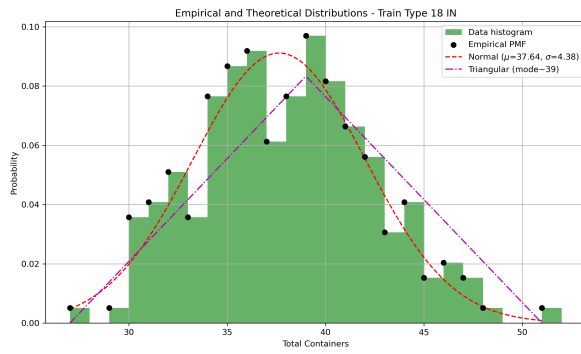


(a) Inbound

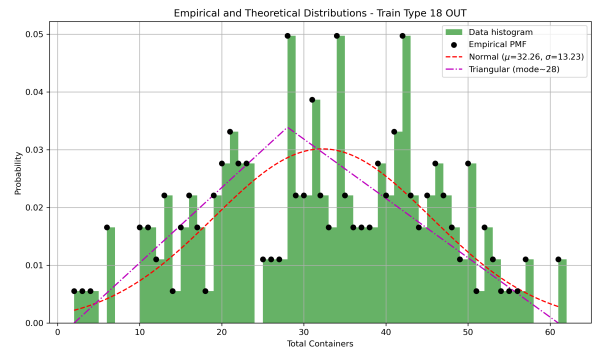


(b) Outbound

Figure 10.14: Distribution of containers for Train Type 16: (a) inbound trains and (b) outbound trains.

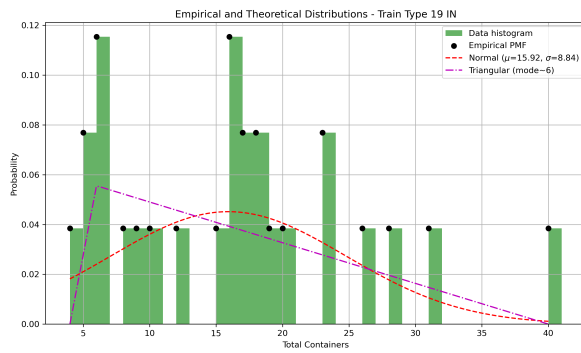


(a) Inbound

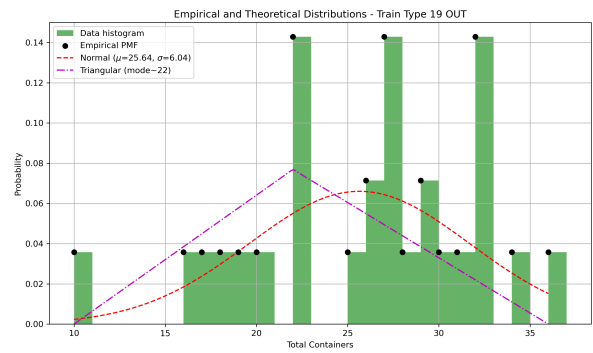


(b) Outbound

Figure 10.15: Distribution of containers for Train Type 18: (a) inbound trains and (b) outbound trains.



(a) Inbound



(b) Outbound

Figure 10.16: Distribution of containers for Train Type 19: (a) inbound trains and (b) outbound trains.

Bibliography

- [1] UNCTAD, “Maritime transport.” <https://unctad.org/statistics>, 2025. Accessed: 2025-07-27.
- [2] P. International, “PSA INTERNATIONAL.” <https://www.globalpsa.com/>, 2026. Accessed: 2026-03-16.
- [3] M. Jamshidi, *Systems of systems engineering: principles and applications*. CRC press, 2017.
- [4] C. Caballini, S. Sacone, and S. Siri, “The port as a system of systems: A system dynamics simulation approach,” in *2012 7th International Conference on System of Systems Engineering (SoSE)*, pp. 191–196, IEEE, 2012.
- [5] Maritimenews, “Shipping Lines Halt Red Sea and Hormuz Transits As Regional Security Crisis Escalates.” <https://www.maritimenews.com/red-sea/shipping-lines-suspend-red-sea-hormuz-transits>, 2026. Accessed: 2026-03-06.
- [6] ContainerManagement, “Strait of hormuz crisis halts gulf container trade as carriers invoke force majeure and washington unveils 20bn insurance backstop.” <https://container-mag.com/2026/03/08/strait-of-hormuz-shipping-crisis-global-trade-disruption/>, 2026. Accessed: 2026-03-26.
- [7] A. Imai, E. Nishimura, and S. Papadimitriou, “The dynamic berth allocation problem for a container port,” *Transportation Research Part B: Methodological*, vol. 35, no. 4, pp. 401–417, 2001.
- [8] Y. Guan and R. K. Cheung, “The berth allocation problem: models and solution methods,” *OR spectrum*, vol. 26, no. 1, pp. 75–92, 2004.
- [9] G. Giallombardo, L. Moccia, M. Salani, and I. Vacca, “Modeling and solving the tactical berth allocation problem,” *Transportation Research Part B: Methodological*, vol. 44, no. 2, pp. 232–245, 2010.

- [10] S.-C. Chang, M.-H. Lin, and J.-F. Tsai, "An optimization approach to berth allocation problems," *Mathematics*, vol. 12, no. 5, p. 753, 2024.
- [11] S. Korekane, T. Nishi, K. Tierney, and Z. Liu, "Neural network assisted branch and bound algorithm for dynamic berth allocation problems," *European Journal of Operational Research*, vol. 319, no. 2, pp. 531–542, 2024.
- [12] N. Makhado, T. Paepae, M. Sejeso, and C. Harley, "Berth allocation and quay crane scheduling in port operations: A systematic review," *Journal of marine science and engineering*, vol. 13, no. 7, p. 1339, 2025.
- [13] M. Sammarra, J.-F. Cordeau, G. Laporte, and M. F. Monaco, "A tabu search heuristic for the quay crane scheduling problem," *Journal of Scheduling*, vol. 10, no. 4, pp. 327–336, 2007.
- [14] M. Wei, J. He, C. Tan, J. Yue, and H. Yu, "Quay crane scheduling with time windows constraints for automated container port," *Ocean & Coastal Management*, vol. 231, p. 106401, 2023.
- [15] Y. Dai, Z. Li, and B. Wang, "Optimizing berth allocation in maritime transportation with quay crane setup times using reinforcement learning," *Journal of Marine Science and Engineering*, vol. 11, no. 5, p. 1025, 2023.
- [16] S. Jo and I. Moon, "A hierarchical reinforcement learning approach for real-time berth allocation and quay crane scheduling," 2025.
- [17] H. J. Carlo, I. F. Vis, and K. J. Roodbergen, "Storage yard operations in container terminals: Literature overview, trends, and research directions," *European journal of operational research*, vol. 235, no. 2, pp. 412–430, 2014.
- [18] H. Yu, Y. Deng, L. Zhang, X. Xiao, and C. Tan, "Yard operations and management in automated container terminals: A review," *Sustainability*, vol. 14, no. 6, p. 3419, 2022.
- [19] J. A. Gunawardhana, H. N. Perera, and A. Thibbotuwawa, "Rule-based dynamic container stacking to optimize yard operations at port terminals," *Maritime Transport Research*, vol. 2, p. 100034, 2021.
- [20] M. F. Monaco, M. Sammarra, and G. Sorrentino, "The terminal-oriented ship stowage planning problem," *European Journal of Operational Research*, vol. 239, no. 1, pp. 256–265, 2014.
- [21] Y. Shen, N. Zhao, M. Xia, and X. Du, "A deep q-learning network for ship stowage planning problem," *Polish Maritime Research*, vol. 24, no. s3, pp. 102–109, 2017.

- [22] P. Legato, R. M. Mazza, and R. Trunfio, "Simulation-based optimization for discharge/loading operations at a maritime container terminal," *OR spectrum*, vol. 32, no. 3, pp. 543–567, 2010.
- [23] S. Siri, A. Palmiere, and D. Ambrosino, "Multi-objective optimization methods for train load planning in seaport container terminals," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 3, pp. 3216–3228, 2023.
- [24] E. Zehendner and D. Feillet, "Benefits of a truck appointment system on the service quality of inland transport modes at a multimodal container terminal," *European Journal of Operational Research*, vol. 235, no. 2, pp. 461–469, 2014.
- [25] W. Zhao and A. V. Goodchild, "Using the truck appointment system to improve yard efficiency in container terminals," *Maritime Economics & Logistics*, vol. 15, no. 1, pp. 101–119, 2013.
- [26] "THE PSA GENOA PRA' TERMINAL." <https://www.psagp.it/>, 2025. Accessed: 2025-10-30.
- [27] Z. Li, L. Wang, and H. Lou, "Analysis and forecasting on the relationship between port logistics capacity and coastal marine economic growth based on pknn in case of zhejiang province," in *2012 Eighth International Conference on Computational Intelligence and Security*, pp. 247–251, IEEE, 2012.
- [28] H.-S. Hwang, S.-T. Bae, and G.-S. Cho, "Container terminal demand forecasting framework using fuzzy-gmdh and neural network method," in *Second International Conference on Innovative Computing, Informatio and Control (ICICIC 2007)*, pp. 119–119, IEEE, 2007.
- [29] G. Fancello, C. Pani, M. Pisano, P. Serra, P. Zuddas, and P. Fadda, "Prediction of arrival times and human resources allocation for container terminal," *Maritime Economics & Logistics*, vol. 13, pp. 142–173, 2011.
- [30] A. Viellechner and S. Spinler, "Novel data analytics meets conventional container shipping: predicting delays by comparing various machine learning algorithms," 2020.
- [31] N. Wang, S. Shen, J. Cao, Y. Ding, and Y. Xiao, "A system for container terminal operation prediction," *2020 IEEE International Conference on Progress in Informatics and Computing (PIC)*, pp. 407–411, 2020.
- [32] Y. Gao, D. Chang, T. Fang, and Y. Fan, "The daily container volumes prediction of storage yard in port with long short-term memory recurrent neural network," *Journal of Advanced Transportation*, vol. 2019, pp. 1–11, 2019.

- [33] Z. Wang, Q. Zeng, and M. Lv, "Modeling the external truck arrivals in container terminals based on dbn and svm," *ICIC Express Letters*, vol. 12, no. 10, pp. 1033–1040, 2018.
- [34] M.-j. Ding, S.-z. Zhang, Z. Hai-dong, Y.-h. Wu, and L.-b. Zhang, "A prediction model of the sum of container based on combined bp neural network and svm," *Journal of Information Processing Systems*, vol. 15, no. 2, pp. 305–319, 2019.
- [35] A. Nadi, S. Sharma, M. Snelder, T. Bakri, H. van Lint, and L. Tavasszy, "Short-term prediction of outbound truck traffic from the exchange of information in logistics hubs: A case study for the port of rotterdam," *Transportation Research Part C: Emerging Technologies*, vol. 127, p. 103111, 2021.
- [36] I. Kourounioti, A. Polydoropoulou, and C. Tsiklidis, "Development of models predicting dwell time of import containers in port container terminals—an artificial neural networks application," *Transportation Research Procedia*, vol. 14, pp. 243–252, 2016.
- [37] F. D. Kanellos, "Multiagent-system-based operation scheduling of large ports' power systems with emissions limitation," *IEEE Systems Journal*, vol. 13, no. 2, pp. 1831–1840, 2019.
- [38] C. Caballini and S. Sacone, "Modeling and simulation of the rail port cycle," *IEEE Systems Journal*, vol. 9, no. 1, pp. 273–282, 2015.
- [39] H. M. Al-Deek, "Use of vessel freight data to forecast heavy truck movements at seaports," *Transportation research record*, vol. 1804, no. 1, pp. 217–224, 2002.
- [40] N. Li, H. Sheng, P. Wang, Y. Jia, Z. Yang, and Z. Jin, "Modeling categorized truck arrivals at ports: Big data for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 2772–2788, 2022.
- [41] N. Li, Z. Wang, X. Lin, and H. Sheng, "Prediction of delivery truck arrivals at container terminals: an ensemble deep learning model," *Maritime Economics & Logistics*, pp. 1–27, 2024.
- [42] C. Caballini and S. Sacone, "Simulation of novel algorithms to reduce truck congestion at container terminals," in *2021 7th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pp. 1–6, IEEE, 2021.
- [43] C. Caballini, J. Mar-Ortiz, M. Gracia, and S. Sacone, "Optimal truck scheduling in a container terminal by using a truck appointment system," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2525–2530, IEEE, 2018.
- [44] A. T. Hoang, A. M. Foley, S. Nižetić, Z. Huang, H. C. Ong, A. I. Ölçer, X. P. Nguyen, *et al.*, "Energy-related approach for reduction of co2 emissions: A critical strategy on the port-to-ship pathway," *Journal of Cleaner Production*, vol. 355, p. 131772, 2022.

- [45] P. Yun, L. Xiangda, W. Wenyuan, L. Ke, and L. Chuan, "A simulation-based research on carbon emission mitigation strategies for green container terminals," *Ocean Engineering*, vol. 163, pp. 288–298, 2018.
- [46] C. Pasquale, S. Liu, S. Siri, S. Sacone, and B. De Schutter, "A new emission model including on-ramps for two-class freeway traffic control," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pp. 1143–1149, IEEE, 2015.
- [47] H. Geerlings and R. Van Duin, "A new method for assessing co2-emissions from container terminals: a promising approach applied in rotterdam," *Journal of cleaner Production*, vol. 19, no. 6-7, pp. 657–666, 2011.
- [48] J. Sim, "A carbon emission evaluation model for a container terminal," *Journal of cleaner production*, vol. 186, pp. 526–533, 2018.
- [49] G. K. Saharidis and G. E. Konstantzos, "Critical overview of emission calculation models in order to evaluate their potential use in estimation of greenhouse gas emissions from in port truck operations," *Journal of Cleaner Production*, vol. 185, pp. 1024–1031, 2018.
- [50] H. Lee, H. T. Pham, C. Kim, and K. Lee, "A study on emissions from drayage trucks in the port city-focusing on the port of incheon," *Sustainability*, vol. 11, no. 19, p. 5358, 2019.
- [51] M. Ma, H. Fan, X. Jiang, and Z. Guo, "Truck arrivals scheduling with vessel dependent time windows to reduce carbon emissions," *Sustainability*, vol. 11, no. 22, p. 6410, 2019.
- [52] G. Chen, K. Govindan, and M. M. Goliass, "Reducing truck emissions at container terminals in a low carbon economy: Proposal of a queueing-based bi-objective model for optimizing truck arrival pattern," *Transportation Research Part E: Logistics and Transportation Review*, vol. 55, pp. 3–22, 2013.
- [53] H.-C. Chu and M. D. Meyer, "Methodology for assessing emission reduction of truck-only toll lanes," *Energy Policy*, vol. 37, no. 8, pp. 3287–3294, 2009.
- [54] Z. Xu, Y. Kang, Y. Cao, and Z. Li, "Deep amended copert model for regional vehicle emission prediction," *Science China Information Sciences*, vol. 64, pp. 1–3, 2021.
- [55] H.-C. L. Fan, W. Wilson, and B. Dahl, "Congestion, port expansion and spatial competition for us container imports," *Transportation Research Part E*, vol. 48, no. 6, pp. 1121–1136, 2012.
- [56] H.-C. H.C. Chen and S. Liu, "Should ports expand their facilities under congestion and uncertainty?," *Transportation Research Part B*, vol. 85, pp. 109–131, 2016.
- [57] X. Wang and Q. Meng, "Optimal price decisions for joint ventures between port operators and shipping lines under the congestion effect," *Transportation Research Part E*, vol. 273, no. 2, pp. 695–707, 2019.

- [58] W. Talley and N. ManWo, "Port multi-service congestion," *Transportation Research Part E*, vol. 94, pp. 66–70, 2016.
- [59] M. Hyland, B.-M. Lama, H. Mahmassani, I. Verbas, X. Xu, K. Smilowitz, and B. Johnson, "Potential for a logistics island to circumvent container port congestion in a constrained environment," *Transport Policy*, vol. 86, pp. 50–59, 2020.
- [60] G. Chen, K. Govindan, and Z. Yang, "Managing truck arrivals with time windows to alleviate gate congestion at container terminals," *International Journal of Production Economics*, vol. 141, no. 1, pp. 179–188, 2013.
- [61] I. Motono, M. Furuichi, T. Ninomiya, S. Suzuki, and M. Fuse, "Insightful observations on trailer queues at landside container terminal gates: What generates congestion at the gates?," *Research in Transportation Business and Management*, vol. 19, pp. 118–131, 2016.
- [62] L. Zhen, "Modeling of yard congestion and optimization of yard template in container ports," *Transportation Research Part B*, vol. 90, pp. 83–104, 2016.
- [63] Y. Huiyun, L. Xin, X. Lixuan, L. Xiangjun, J. Zhihong, and B. Zhan, "Truck appointment at container terminals: Status and perspectives," in *2018 Chinese Control And Decision Conference (CCDC)*, pp. 1954–1960, IEEE, 2018.
- [64] C. Caballini, M. D. Gracia, J. Mar-Ortiz, and S. Saccone, "A combined data mining–optimization approach to manage trucks operations in container terminals with the use of a tas: Application to an italian and a mexican port," *Transportation Research Part E*, vol. 142, 2020.
- [65] A. M. Abdelmagid, M. S. Gheith, and A. B. Eltawil, "A comprehensive review of the truck appointment scheduling models and directions for future research," *Transport Reviews*, vol. 42, no. 1, 2022.
- [66] K. Chargui, T. Zouadi, A. El Fallahi, M. Reghioui, and T. Aouam, "Berth and quay crane allocation and scheduling with worker performance variability and yard truck deployment in container terminals," *Transportation Research Part E: Logistics and Transportation Review*, vol. 154, p. 102449, 2021.
- [67] S. Amirifar, A. Tavakoli Kashani, and E. Hassannayebi, "Optimal berth planning in container terminals: A novel algorithm for quay crane scheduling," *Quarterly Journal of Transportation Engineering*, 2025.
- [68] Y. Chang, X. Zhu, B. Yan, and L. Wang, "Integrated scheduling of handling operations in railway container terminals," *Transportation Letters*, vol. 11, no. 7, pp. 402–412, 2019.

- [69] T. Xia, L. Wang, Q. Zhang, J.-X. Dong, D.-P. Song, and X. Zhu, “Integrated scheduling of gantry cranes, container trucks and yard cranes in on-dock railway operation areas at multimodal container ports,” *IET Intelligent Transport Systems*, vol. 19, no. 1, p. e12600, 2025.
- [70] Y. Xie and D.-P. Song, “Optimal planning for container prestaging, discharging, and loading processes at seaport rail terminals with uncertainty,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 119, pp. 88–109, 2018.
- [71] H. Y. Lau and Y. Zhao, “Integrated scheduling of handling equipment at automated container terminals,” *International journal of production economics*, vol. 112, no. 2, pp. 665–682, 2008.
- [72] J. Luo, Y. Wu, and A. B. Mendes, “Modelling of integrated vehicle scheduling and container storage problems in unloading process at an automated container terminal,” *Computers & Industrial Engineering*, vol. 94, pp. 32–44, 2016.
- [73] B. Xu, D. Jie, J. Li, Y. Yang, F. Wen, and H. Song, “Integrated scheduling optimization of u-shaped automated container terminal under loading and unloading mode,” *Computers & Industrial Engineering*, vol. 162, p. 107695, 2021.
- [74] M. Gendreau and J.-Y. Potvin, “Tabu search,” in *Search methodologies: introductory tutorials in optimization and decision support techniques*, pp. 243–263, Springer, 2013.
- [75] C. Bierwirth and F. Meisel, “A follow-up survey of berth allocation and quay crane scheduling problems in container terminals,” *European Journal of Operational Research*, vol. 244, no. 3, pp. 675–689, 2015.
- [76] S. Voß, “Meta-heuristics: The state of the art,” in *Workshop on Local Search for Planning and Scheduling*, pp. 1–23, Springer, 2000.
- [77] Z. Feng, C. Cao, Y. Liu, and Y. Zhou, “A multiobjective optimization for train routing at the high-speed railway station based on tabu search algorithm,” *Mathematical Problems in Engineering*, vol. 2018, no. 1, p. 8394397, 2018.
- [78] Q. Hu, F. Corman, B. Wiegman, and G. Lodewijks, “A tabu search algorithm to solve the integrated planning of container on an inter-terminal network connected with a hinterland rail network,” *Transportation Research Part C: Emerging Technologies*, vol. 91, pp. 15–36, 2018.
- [79] B. Skinner, S. Yuan, S. Huang, D. Liu, B. Cai, G. Dissanayake, H. Lau, A. Bott, and D. Pagac, “Optimisation for job scheduling at automated container terminals using genetic algorithm,” *Computers & Industrial Engineering*, vol. 64, no. 1, pp. 511–523, 2013.

- [80] S. M. Homayouni, S. H. Tang, and O. Motlagh, "A genetic algorithm for optimization of integrated scheduling of cranes, vehicles, and storage platforms at automated container terminals," *Journal of Computational and Applied Mathematics*, vol. 270, pp. 545–556, 2014.
- [81] S. Homayouni, S. Tang, N. Ismail, and M. Ariffin, "Using simulated annealing algorithm for optimization of quay cranes and automated guided vehicles scheduling," *International Journal of the Physical Sciences*, vol. 6, no. 27, pp. 6286–6294, 2011.
- [82] C. Caballini, S. Fioribello, S. io, and S. Siri, "An milp optimization problem for sizing port rail networks and planning shunting operations in container terminals," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 4, pp. 1492–1503, 2016.
- [83] R. L. Burdett, "Multi-objective models and techniques for analysing the absolute capacity of railway networks," *European Journal of Operational Research*, vol. 245, no. 2, pp. 489–505, 2015.
- [84] T. Liu and H. Wang, "Evaluating the service capacity of port-centric intermodal transshipment hub," *Journal of Marine Science and Engineering*, vol. 11, no. 7, p. 1403, 2023.
- [85] M. N. Abourraja, N. Kringos, and S. Meijer, "Exploiting simulation model potential in investigating handling capacity of ro-ro terminals: The case study of norvik seaport," *Simulation Modelling Practice and Theory*, vol. 117, p. 102513, 2022.
- [86] C. Caballini, C. Pasquale, S. io, and S. Siri, "An event-triggered receding-horizon scheme for planning rail operations in maritime terminals," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 365–375, 2013.
- [87] M. Marinov and J. Viegas, "A simulation modelling methodology for evaluating flat-shunted yard operations," *Simulation Modelling Practice and Theory*, vol. 17, no. 6, pp. 1106–1129, 2009.
- [88] A. García and I. García, "A simulation-based flexible platform for the design and evaluation of rail service infrastructures," *Simulation Modelling Practice and Theory*, vol. 27, pp. 31–46, 2012.
- [89] Y.-C. Lai and C. P. Barkan, "Enhanced parametric railway capacity evaluation tool," *Transportation research record*, vol. 2117, no. 1, pp. 33–40, 2009.
- [90] V. Reis and A. Almeida, "Capacity evaluation of a railway terminal using microsimulation: Case study of a freight village in turin," *Frontiers in Built Environment*, vol. 5, p. 75, 2019.
- [91] M. Sadeghi, M. Bagheri, and M. S. Pishvaei, "Evaluation of rail terminals in container ports using simulation: a case study," *Simulation*, vol. 97, no. 12, pp. 809–820, 2021.

- [92] P. Srisurin, P. Pimpanit, and P. Jarumaneeroj, "Evaluating the long-term operational performance of a large-scale inland terminal: A discrete event simulation-based modeling approach," *Plos One*, vol. 17, no. 12, p. e0278649, 2022.
- [93] E. Bergeron, J.-F. Audy, and P. Forget, "Analysis of a methodology for simulating a port logistics system to evaluate rail capacity in bulk ports," *Transportation Research Procedia*, vol. 82, pp. 3690–3709, 2025.
- [94] A. E. Rizzoli, N. Fornara, and L. M. Gambardella, "A simulation tool for combined rail/road transport in intermodal terminals," *Mathematics and computers in simulation*, vol. 59, no. 1-3, pp. 57–71, 2002.
- [95] B. Dragović, E. Tzannatos, and N. K. Park, "Simulation modelling in ports and container terminals: literature overview and analysis by research field, application area and tool," *Flexible Services and Manufacturing Journal*, vol. 29, no. 1, pp. 4–34, 2017.
- [96] V. Kumar and M. Garg, "Predictive analytics: a review of trends and techniques," *International Journal of Computer Applications*, vol. 182, no. 1, pp. 31–37, 2018.
- [97] K. Gurney, *An introduction to neural networks*. CRC press, 2018.
- [98] M. Cilimkovic, "Neural networks and back propagation algorithm," *Institute of Technology Blanchardstown, Blanchardstown Road North Dublin*, vol. 15, no. 1, 2015.
- [99] H. K. Cigizoglu and Ö. Kişi, "Flow prediction by three back propagation techniques using k-fold partitioning of neural network training data," *Hydrology Research*, vol. 36, no. 1, pp. 49–64, 2005.
- [100] S. Radhimeenakshi and G. Nasira, "Prediction of heart disease using neural network with back propagation," *International Journal of Data Mining Techniques and Applications*, vol. 4, no. 1, pp. 19–22, 2015.
- [101] M. J. Rana, M. S. Shahriar, and M. Shafiullah, "Levenberg–marquardt neural network to estimate upfc-coordinated pss parameters to enhance power system stability," *Neural Computing and Applications*, vol. 31, pp. 1237–1248, 2019.
- [102] M. Rafał, "Cross validation methods: Analysis based on diagnostics of thyroid cancer metastasis," *ICT Express*, vol. 8, no. 2, pp. 183–188, 2022.
- [103] T. Hastie, "The elements of statistical learning: data mining, inference, and prediction," 2009.
- [104] L. A. Yates, Z. Aandahl, S. A. Richards, and B. W. Brook, "Cross validation for model selection: a review with examples from ecology," *Ecological Monographs*, vol. 93, no. 1, p. e1557, 2023.

- [105] L. Ntziachristos and Z. Samaras, “1.a.3.b.i-iv road transport 2023 final,” 2023.
- [106] G. C. Calafiore and L. El Ghaoui, *Optimization models*. Cambridge university press, 2014.
- [107] P. Festa, “A brief introduction to exact, approximation, and heuristic algorithms for solving hard combinatorial optimization problems,” in *2014 16th International Conference on Transparent Optical Networks (ICTON)*, pp. 1–20, IEEE, 2014.
- [108] F. S. Hillier, *Introduction to operations research*. McGrawHill, 2005.
- [109] M. L. Pinedo, *Scheduling*, vol. 29. Springer, 2012.
- [110] M. L. Pinedo, *Planning and scheduling in manufacturing and services*. Springer, 2005.
- [111] A. M. Law, W. D. Kelton, and W. D. Kelton, *Simulation modeling and analysis*, vol. 3. Mcgraw-hill New York, 2007.
- [112] M. Beaverstock, A. Greenwood, E. Lavery, and W. Nordgren, “Applied simulation,” *Modeling and Analysis using Flexsim, Flexsim Software Products, Inc., Canyon Park Technology Center, Orem, USA*, 2011.
- [113] FlexSim, “FlexSim.” <https://docs.flexsim.com>, 2026. Accessed: 2026-01-11.
- [114] FlexTerm, *FlexTerm User Documentation*. FlexTerm, 2024. FlexTerm Help online, Version 3.0, accessed via local software installation; Copyright © 2012–2024.
- [115] FlexTerm, *FlexTerm Travel Manager User Manual*. FlexTerm, 2024. FlexTerm Help online, Version 3.0, accessed via local software installation; Copyright © 2012–2024.
- [116] R. Hoxha, A. Olmo, C. Pasquale, R. Sacile, S. Sacone, and E. Zero, “Neural network-based prediction of import container outflows via trucks: A system-of-systems analysis in port terminals,” in *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*, pp. 567–572, IEEE, 2024.
- [117] R. Hoxha, S. Sacone, S. Cangelosi, and E. Zero, “Truck arrival optimization for import container pickup at port terminal areas to reduce congestion,” in *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2850–2855, IEEE, 2024.
- [118] R. Hoxha, S. Cangelosi, S. Sacone, and E. Zero, “Truck emission forecasting and peak mitigation in a port terminal,” *IFAC-PapersOnLine*, vol. 58, no. 2, pp. 192–198, 2024.
- [119] R. Hoxha, S. Cangelosi, S. Sacone, and E. Zero, “Reducing emission peaks in a port terminal through optimized truck arrivals,” *IFAC-PapersOnLine*, vol. 58, no. 10, pp. 321–326, 2024.

- [120] Y. Du, J. Li, C. Li, and P. Duan, “A reinforcement learning approach for flexible job shop scheduling problem with crane transportation and setup times,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 4, pp. 5695–5709, 2022.
- [121] A. Bozzi, S. Graffione, R. Sacile, and E. Zero, “Dynamic mpc-based scheduling in a smart manufacturing system problem,” *IEEE Access*, 2023.
- [122] A. Bozzi, S. Graffione, J. Jiménez, R. Sacile, and E. Zero, “Reliability evaluation of emergent behaviour in a flexible manufacturing problem,” in *2023 18th Annual System of Systems Engineering Conference (SoSe)*, pp. 1–7, IEEE, 2023.
- [123] R. Hoxha, A. Bozzi, S. Cangelosi, S. Sacone, and E. Zero, “Optimizing train discharge in port terminals: A flexible flow-shop scheduling approach,” in *2025 20th Annual System of Systems Engineering Conference (SoSe)*, pp. 1–6, IEEE, 2025.
- [124] R. Hoxha, K. Kumbria, J. Carlier, E. Zero, S. Cangelosi, and S. Sacone, “Efficient rail discharge scheduling in port terminals: A tabu search approach,” in *2026 IEEE 29th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2026. Accepted for publication.
- [125] R. Hoxha, S. Sacone, G. Van de Weyer, R. De Ridder, S. Cangelosi, and E. Zero, “Evaluating rail yard capacity with an optimization model,” in *2025 IEEE 28th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 27–32, IEEE, 2025.
- [126] R. Hoxha, R. De Ridder, G. Van de Weyer, S. Cangelosi, S. Sacone, and E. Zero, “A system-of-systems simulation approach to rail yard capacity evaluation in a port terminal,” in *2025 International Congress on Smart Agriculture and Sustainable Systems (SmartAgri&SuSY)*, pp. 1–5, IEEE, 2025.