UNIVERSITY OF MODENA AND REGGIO EMILIA
Department of Sciences and Methods for Engineering
Doctorate School in Industrial Innovation Engineering
XXXVIII Cycle

Data-Driven Optimization of Intralogistics in the Ceramic Tile Industry

DOCTORAL THESIS

Author:
Marco TACCINI

Supervisor:
Prof. Manuel IORI
Co-Supervisor:
Prof. Anand SUBRAMANIAN

Coordinator:
Prof. Franco ZAMBONELLI

UNIVERSITY OF MODENA AND REGGIO EMILIA

Abstract

Doctorate School in Industrial Innovation Engineering
Department of Sciences and Methods for Engineering

Doctor of Philosophy

Data-Driven Optimization of Intralogistics in the Ceramic Tile Industry

by Marco TACCINI

This doctoral thesis explores optimization problems in the field of intralogistics, with a particular focus on the ceramic tile industry. The study focuses on three main research directions aimed at improving the efficiency, flexibility, and reliability of intralogistics processes through the application of Operations Research (OR), Discrete Event Simulation (DES) and Machine Learning (ML) methodologies. The first topic concerns the optimization of buffer storage areas by means of DES. The research analyzes different pallet classification and allocation policies, providing quantitative insights to maximize storage capacity, facilitate operators' activities, and enhance warehouse efficiency. The second topic addresses the design of an automated Decision Support System (DSS) for transportation planning in peripheral warehouses. The system integrates digitalization and mathematical optimization to automate order assignments and truck loading plans, minimizing transportation costs and improving coordination among storage sites. The third topic explores a hybrid predictive approach that combines optimization and ML to estimate the number of pallets required for shipping. The algorithm enables fast and accurate predictions suitable for real-time decision-making and is further extended through a multi-output regression framework capable of simultaneously forecasting different pallet types, improving stock planning and resource utilization. In conclusion, this doctoral thesis contributes to the advancement of OR applications in ceramic tiles intralogistics, providing innovative methodologies and practical tools that support the digitalization and optimization of industrial logistics systems.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In the current economic and technological landscape, efficient management of logistics processes represents a strategic factor for companies operating in highly competitive sectors. Especially, the ceramic tile industry, characterized by large production of heavy, voluminous and fragile products, requires advanced methodologies to ensure flexibility, reliability, and cost efficiency in intralogistics operations. Operations Research (OR) offers a wide range of tools to address these challenges. By combining mathematical modeling, algorithmic approaches, and data-driven techniques, it enables the design of solutions that improve decision-making and operational efficiency.

In recent years, the ceramic tile sector has been facing a set of challenges that further increase the complexity of intralogistics management. These include increasing market uncertainty, a widening product range, shorter delivery times, and growing pressure to reduce operational costs while maintaining high quality standards. Moreover, the rapid evolution of automation technologies and digital platforms has significantly transformed intralogistics processes. In this dynamic context, traditional experience-based and manual approaches, which are still widely adopted in the ceramic tile sector, may be inadequate to effectively manage the scale, heterogeneity, and uncertainty that characterize modern intralogistics. Addressing these challenges can benefit from a multidisciplinary viewpoint that brings together methods and competencies from OR, industrial engineering, and data analytics. In this sense, this thesis explores a multidisciplinary approach that combines different methodological paradigms with the aim of improving and optimizing intralogistics operations in the ceramic tile sector.

This doctoral research is developed along three main directions. The first investigates the use of Discrete Event Simulation (DES) to study and optimize the placement of pallets within stock areas, aiming to maximize storage capacity, facilitate operators' activities, and improve warehouse performance. The second concerns the development of an automated Decision Support System (DSS) for optimizing transportation activities in peripheral warehouses, with the goal of minimizing costs and enhancing the efficiency of flows between storage locations. Finally, the third examines hybrid approaches that integrate optimization and Machine Learning (ML) techniques to predict the number of shipping pallets, supporting proactive decision-making and ensuring more efficient logistics flows. Together, these contributions advance both the theoretical understanding and practical application of OR in intralogistics, providing insights that can be directly implemented in industrial settings.

The motivation for the present study arises from the growing interest within the scientific community in addressing the challenges of intralogistics optimization, particularly in manufacturing sectors, like ceramic tile industry, where efficiency and flexibility are critical. In recent years, a significant body of research has emerged on the use of simulation, optimization, and data-driven methods to enhance logistics performance (Aretoulaki et al., 2024), reflecting the relevance and timeliness of the themes explored in this thesis.

Another driving factor behind this research stems from the interest cultivated throughout my industrial doctorate in an international ceramic tile company headquartered in Italy, located in the ceramic district of Sassuolo (*Confindustria Ceramica Sassuolo* 2025). Within this context, I had the opportunity to work directly on projects aligned with the company operational needs, applying the methodologies and techniques developed during the doctoral research to real-world problems. The outcomes of this collaboration were not only disseminated through scientific publications but also implemented within the company logistics processes, providing both economic and practical benefits. This industrial doctorate has provided a unique opportunity to bridge the gap between academic research and industrial application. It allowed for the validation of theoretical models through practical deployment, while at the same time contributing to the innovation and competitiveness of a key sector within the Italian manufacturing landscape. The work presented in this thesis reflects this dual perspective, combining rigorous academic development with direct applicability to industrial practice.

More in detail, the thesis is organized into four main chapters. Chapter 2 investigates the use of DES to study and optimize the management of buffer storage areas in the ceramic tile industry. The motivation for this work arises from several distinctive characteristics of ceramic production such as the high lack of homogeneity in terms of size, finishing, and packaging configuration (Boza et al., 2014). As a result, the pallets that are transported and stored within the factory are extremely heterogeneous, which poses a challenge for the efficient organization of buffer areas. Traditional allocation strategies often fail to guarantee an optimal use of space, leading to wasted capacity, operators inefficiency and higher risks of product damage. In this context, buffer areas play a crucial role, acting as temporary storage zones between production lines and the shipping warehouse. The way pallets are classified and positioned within these areas strongly affects the overall logistics performance. A suboptimal allocation strategy can increase the workload of forklift operators, extend the travel distance within the warehouse, and create congestion, with negative consequences on productivity and safety. At the same time, companies face the additional constraint of limited storage capacity due to the large volumes produced daily, which makes the efficient use of buffer space a priority. To address these challenges, DES is applied as a decision-support methodology. It enables the evaluation of different pallet allocation and classification policies before their actual implementation with the objective of identifying strategies that maximize storage capacity, facilitate operators' activities, and enhance warehouse efficiency. In particular, the research compares several alternative allocation rules, highlighting trade-offs between space utilization and operational complexity. The contribution of this chapter shows how simulation can capture the dynamics of highly heterogeneous storage systems and provide quantitative insights to support intralogistics decision-making in the ceramic tile industry.

Chapter 3 presents the development of an automated DSS for the planning and optimization of transportation activities in peripheral warehouses of a ceramic tile company. The motivation for this research comes from the traditional practices still widely adopted in the sector, where order assignments and truck loading plans are often managed manually by logistics operators. This manual approach is not only time-consuming and error-prone, but it also generates significant inefficiencies: orders are processed with limited visibility of the overall flows and redundant activities are performed. The ceramic tile industry amplifies these challenges due to its complex logistics structure. Large volumes of different products need to be shipped daily from multiple peripheral storage locations to a central distribution hub, where shipments are consolidated. Pallets are destined for different locations, while demand fluctuations and last-minute changes in customer orders add uncertainty to the planning process. In this context, relying on manual procedures leads to suboptimal truck utilization and high transportation costs. To address these issues, the research proposes a DSS that integrates digitalization with mathematical optimization. The system digitalizes the order management process, eliminating manual-based and repetitive activities, and incorporates an Integer Linear Programming (ILP) model designed to optimize the assignment of orders to warehouses and trucks. The model minimizes total transportation costs while respecting operational constraints, such as truck capacities, customer delivery requirements, and the synchronization of flows from different warehouses to the central hub. The DSS architecture has been designed for usability and scalability. On the one hand, it ensures that operators can interact with the system through an intuitive interface that provides decision support in real time. On the other hand, the optimization model guarantees that solutions are systematically cost-efficient and consistent with company objectives. By automating the planning process, the DSS reduces the workload of logistics operators, increases the reliability of decisions, and improves coordination across multiple peripheral warehouses. The contribution of this chapter lies in demonstrating how OR models can be embedded into decision-support platforms that are directly applicable in industrial contexts. From a practical perspective, the implementation of the DSS led to measurable benefits, including reductions in transportation costs and simplification of order management processes.

Chapter 4 presents the development of a hybrid approach that combines ML with optimization to quickly and accurately predict the number of pallets required to ship an order of ceramic tiles. The motivation for this research stems from a real-world case study in ceramic tile production, where the company needs to provide customers with an immediate and accurate expectation of transportation costs at the moment of purchase on its e-commerce platform. Furthermore, the operational reality of the company, which includes warehouse uncertainties and loading operations primarily based on the personal experience of operators, means that traditional model-based solution methods are not well suited. The logistics challenges in the ceramic tile industry amplify the difficulty of the problem, which is a variant of the Distributor Pallet Loading Problem (DPLP) (Mungwattana et al., 2023). Loading operations of shipping pallets must satisfy complex requirements, including weight and volume constraints, vertical stability, load-bearing, and restricted box orientations. Moreover, due to the warehouse current state and system decisions, products are not all available simultaneously. In this context, the objective shifts from finding a time-consuming optimal solution to providing a quick and reliable prediction. To address these issues, the research proposes a hybrid algorithm where an ML model, specifically XGBoost (Chen and Guestrin, 2016a), is trained over a company dataset comprising two years of customer orders. The predictive accuracy of the ML model is significantly improved by incorporating additional features derived from quick optimization algorithms. These optimization-based features include a linear relaxation lower bound, an upper bound calculated assuming homogeneous pallets, and the result from a quick constructive heuristic. The contribution of this chapter lies in demonstrating the effectiveness of a hybrid approach for complex combinatorial optimization problems, especially those with company-specific operational constraints that are difficult to model explicitly.

Chapter 5 presents an extension of the hybrid approach developed in Chapter 4, focusing on the implementation of a Multi-Output Regression (MOR) technique to further enhance the predictive capabilities for the DPLP variant. The core motivation remains the company critical need for fast and accurate predictions of the number of pallets required for shipping to facilitate quick cost estimation on their e-commerce platform and manage operational workflows. The challenge addressed in this extension is related to the different types of shipping pallet available. While the algorithm developed in Chapter 4 successfully predicts the total number of pallets required, this work allows the generalization of the model for multi-output predictions, specifically for small-sized and big-sized pallets simultaneously. This is crucial because the choice between pallet sizes is a key operational and cost-driver decision. Starting from the algorithm introduced in Chapter 4, two main MOR strategies are investigated. The first is Independent Regressors (IR), a baseline approach that considers the prediction of quantities for each pallet type as separate and uncorrelated tasks, training an individual regressor for each output. The second is the Regressor Chain (RC), a more advanced method that explicitly captures dependencies between outputs. It trains a first regressor to predict one target, then incorporates this prediction as an additional input feature for the subsequent regressor, which estimates the next target. Two variants of the chain were evaluated, differing in the order of the predicted outputs. The contribution of this chapter shows that adopting a MOR framework is well suited to the operational setting, where different pallet types are inherently correlated. Moreover, by forecasting the quantities of small and large pallets, the system supports the company in improving stock planning and optimizing the use of warehouse and transportation resources.

Finally, concluding remarks are reported in Chapter 6.

References

Aretoulaki, Eleni et al. (2024). "Discrete event simulation and Digital Twins in warehouse logistics: a bibliometric and content analysis-based systematic literature review". In: *International Journal of Computer Integrated Manufacturing* 37.10-11, pp. 1376–1403.

Boza, Andrés et al. (2014). "A Model-Driven DSS Architecture for Delivery Management in Collaborative Supply Chains With Lack of Homogeneity in Products". In: *Production Planning & Control* 25.8, pp. 650–661.

Chen, Tianqi and Carlos Guestrin (2016a). "XGBoost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.

Confindustria Ceramica Sassuolo (2025). URL: https://confindustriaceramica.it/ (visited on 09/2025).

Mungwattana, Anan, Thanakrit Piyachayawat, and Gerrit K. Janssens (2023). "A two-step evolutionary algorithm for the distributor's pallet loading problem with multi-size pallets". In: *Flexible Services and Manufacturing Journal* 35.4, pp. 1256–1275.

Chapter 2

Simulation in Ceramic Tile Industry [1]

In this chapter we present the study on identifying the best strategy to temporarily store products within a buffer area in an Italian ceramic tile company. The storage policy is analyzed to maximize the storage capacity, facilitate operators' activities, and, consequently, improve the warehouse logistics performance. A DES was conducted using Salabim, a Python based open-source software, in order to determine the best policy. We compare the performance of the current storage policy, based on technical production properties of products, and a newly proposed one, based on products' downstream destination. The results suggested that the proposed strategy significantly improves the performance of the buffer area management. The approach can be applied to different applications, contributing to the literature on simulation-based decision-making in material management. Furthermore, the study provides a functional case study showing the potential and achievable results of Salabim for modeling complex systems.

2.1 Introduction

Over the last decade, the worldwide tile market has experienced increasing significance and remarkable growth. In particular, as reported by ACIMAC Research Department (2022), the global production of tiles grew in general by 7.2% in 2021 compared to the previous year, producing 18.3 billion square meters. Focusing more on Italy, the country is the main world exporter in terms of value, accounting for 28.1% of the world export share, and ranking seventh in terms of production. Furthermore, Baraldi (2022) reveals that Italy produced 435 million square meters of tiles, generating revenues of €6.2 billion. Overall, in 2021, the Italian ceramic sector revealed a positive trend across all key indicators, highlighting its importance.

One of the crucial problems in ceramic industry operations is material handling. As defined in Efthymiou and Ponis (2019), material handling involves a range of activities to move, store, protect and control the products throughout the entire manufacturing process. In particular, products are briefly paused and handled in storage areas, consuming valuable space and time in the form of person-hours and resulting in additional expenses for the company. Therefore, warehouse management, specifically material handling, plays a crucial role in ensuring efficiency and customer satisfaction within the supply chain.

Furthermore, Boza et al. (2014) introduced the concept of Lack of Homogeneity in the Product (LHP), a phenomenon that occurs during production processes when they involve uncertainty, such as natural raw materials or variable operations. The ceramic industry is particularly affected by LHP, due to clays and the highly variable production process, depending on various factors, such as temperature and humidity. The same authors pointed out that LHP is in contrast to customer demands, which in turn require uniform ceramic tiles

[1] The results of this chapter appears in: Taccini, M., Dotti, G., Iori, M. and Subramanian, A. "Improving Buffer Storage Performance in Ceramic Tile Industry via Simulation". In *Proceeding of the International Winter Simulation Conference 2023, pp. 1676-1687.*

for aesthetic reasons. To address this issue, tiles companies must incorporate a classification stage in their production process, involving personalized classification criteria. In particular, in most ceramic industries every piece is visually inspected and classified based on quality, shade, and caliber. Products are then stored in homogeneous subgroups to facilitate their retrieval for homogeneous orders. As a consequence, the classification process can affect material handling performance and, therefore, must be carefully evaluated. Hence, given the importance of material handling in ensuring material flow and avoiding bottlenecks, it is crucial to prevent production stops. Therefore, the objective of this study was to implement a Discrete Event Simulation (DES) in an Italian ceramic tile company, in order to analyze the best storage policy without affecting the material flow. Salabim (Ham (2018)) was selected as simulation tool, as it offers a range of attractive features, including being an open-source Python-based software, having comprehensive documentation, object-oriented architecture, and animations. This work focused specifically on the comparison of two different storage policies for the classification of products within the buffer storage area. In particular, the newly devised policy classifies products based on their downstream destination rather than their production properties, consequently improving the logistic performance in terms of flexibility and costs.

The remainder of the chapter is structured as follows. Section 2.2 provides an overview of the main research areas within the ceramic tile industry, with a particular focus on the applications of DES in the sector. In addition, a review of Salabim and its applications is provided. Section 2.3 presents more details about the problem, while Section 2.4 elaborates the proposed storage policy. Section 2.5 discusses in detail the simulation of the policy, and Section 2.6 reports and comments the results. Finally, Section 2.7 summarizes the study and presents future research directions.

2.2 Literature Review

The ceramic tile industry has received considerable attention in the literature, with many studies exploring different aspects of the industry. A large body of literature has investigated the production process of ceramic tiles. For example, Dubey and Yadava (2008) reviewed the Laser Beam Machine technology to improve the ceramic performance, while Sciancalepore et al. (2014) reviewed antibacterial and self-cleaning coatings for tiles. Moreover, Xie (2008) and Zhao (2021) examined innovative methods to identify tiles defects. More recently, the focus has shifted to the sustainability of the ceramic industry. Ferrari et al. (2019), Medina-Salgado et al. (2021), and Atılgan Türkmen et al. (2021) studied the estimation of the environmental impact of the ceramic industry. Focusing likewise on recycling, Andreola et al. (2016) and Zanelli et al. (2021) examined the usage of recycled materials in tile production, while Mangi et al. (2022) investigated the utilization of the ceramic waste for the production of concrete. Additionally, logistics aspects were studied from the sustainability perspective, as Dondi et al. (2021) highlighted the impact of the supply chain on the ceramic industry sustainability, focusing on identifying the potential criticality in the current consumption trends of raw materials. Moreover, Ma et al. (2022) pursued the achievement of a smart and sustainable manufacturing through the integration of big data analytics and digital twin technologies. The chapter specifically focused on energy-intensive manufacturing companies and provided two different case studies on ceramic tile industries.

Furthermore, a growing body of literature has investigated Decision Support Systems (DSS) applications in various fields within the ceramic tile industry, such as the inventory control in Abdolazimi et al. (2021), and the production planning and scheduling in Soares et al. (2022). In particular, Boza et al. (2014) and Alemany et al. (2018) specifically focused on the delivery management of products with homogeneity requirements, highlighting a crucial aspect of the ceramic tile sector.

Although considerable research has been done on different aspects of the ceramic industry, much less is known about the application of simulation in this sector. To the best of our knowledge, only a few papers investigated process simulation tools as performance measurement tools for improvement in ceramic tile industries. Davoli et al. (2010) developed a simulation tool to evaluate the impact of unreliable orders on the performance of a generic ceramic tile industry, quantified as earnings and fulfilled orders. The stochastic simulation divided the overall model into sub-processes, each simulated in VirtES, a customized simulation software coded in the Scilab environment. A precedent work of Davoli et al. (2008) studied the industrial processes of ceramic tiles manufacturing through the integration of VirtES and AutoMode. Firstly, each industrial process was schematized in terms of input and output and simulated in VirtES, to investigate critical processes and support the process redesign. Then, the impact of the proposed changes was quantified in terms of production, using the AutoMode commercial process-oriented simulation tool. Similarly, Nadir et al. (2020) studied the waste generated by non-efficient tile handling systems, simulating the process in commercial Simio software and comparing the results with performance indicators. However, to the best of our knowledge, no studies have simulated ceramic processes using non-commercial software tools. Therefore, we filled the literature gap by studying the storage policy of a ceramic company through the application of simulation, in particular DES on Salabim, an open-source package developed in Python.

Salabim was presented by Ham (2018), who focused on the characteristics that differentiate Salabim from SimPy, the other open-source simulation software developed in Python. Firstly, Salabim uses the Simula activate/passivate/hold paradigm, which facilitates the implementation of clear and easy-to-maintain models. The Simula approach, introduced by Dahl and Nygaard (1966), enables the modeling of interactions among simulation entities by defining events and changes of state. Moreover, Salabim provides various additional and useful functions such as animation, queues, states, monitors for data collection and presentation, tracing, and statistical distributions.

However, despite its promising characteristics, we were able to find little literature about Salabim. In Lang et al. (2021c), a problem case was used to compare five different DES software tools. In particular, the authors considered three open software tools, including Salabim, and two commercial ones. The results proved that open-source simulation software is a concrete alternative to commercial DES software. Although not first in the ranking, Salabim emerged as a promising tool for DES, particularly recommended to integrate models in Python applications. Several attempts have been made to integrate Machine Learning and DES in Salabim to solve different problems. The main area of interest within the Salabim applications is the simulation of variants of the production scheduling problem, as in Erden et al. (2019), Erden et al. (2021), Lang et al. (2020a), Lang et al. (2020b), Lang et al. (2021a), and Lang et al. (2021b). Furthermore, in Anglano et al. (2019a), and Anglano et al. (2019b), the authors focused on algorithms for the scheduling of applications on a set of heterogeneous mobile devices. Finally, Salabim was also used to simulate operations in health centers, as in Baldwa et al. (2020) and Shoaib and Ramamohan (2022).

Considering the literature gap, this chapter aims to provide a concrete application of DES in the logistics of the ceramic tiles industry. Thus, in the following chapters, we will focus on the Salabim application to the case study.

2.3 Problem Description

This research concerns a international ceramic company that produces tiles according to the make-to-stock strategy. The study first simulates the policy adopted to classify products in the buffer storage area between the production plant and the logistics department. The storage policy is then redesigned as a strategy to maximize the storage capacity, facilitate operators' activities, and, consequently, improve the warehouse logistics performance.

Buffer areas in the ceramic sector are commonly standardized and regularly shaped and therefore they can be schematized as a matrix, as shown in Figure 2.1. Each cell of the matrix can contain a pile of four identical-sized pallets, vertically stacked. The pallets are transported by automatic vehicles into the storage area from the east side (in the bottom of Figure 2.1) and are temporarily stored in the buffer, waiting for the forklift operators on the west side to manually pick them up and finally pack them into the logistics department. Since operators pick pallets manually from the buffer area, their work efficiency is impacted by the location of the pallets. Consequently, the storage policy must carefully assign pallets to cells, dividing pallets into classes based on criteria to cluster them and facilitate picking activities.

To facilitate workers activities, each column of the matrix is dedicated to a single class. Specifically, pallets are deposited into the westernmost available cell of a column dedicated to their class. In case all columns have already been destined to other classes, the system can store distinct classes in the same column, to prevent production downtime. However, mixing classes causes penalties because it hampers operators' activities. Classification policies are described in detail in Section 2.4.

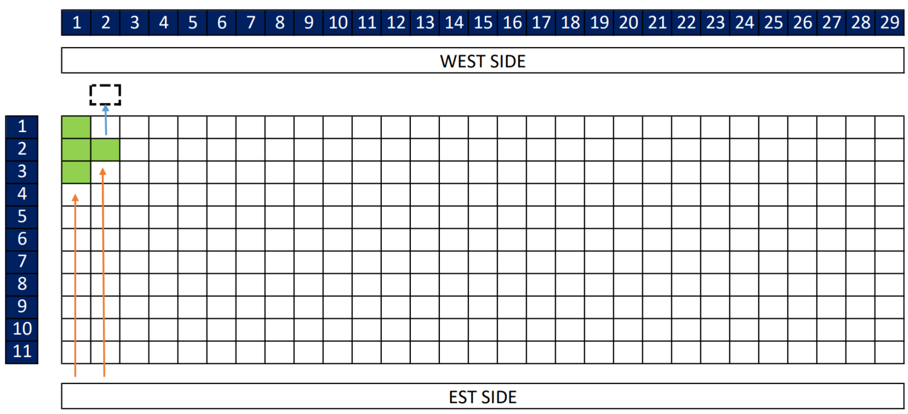


FIGURE 2.1: Storage area representation.

2.4 Classification Policies

2.4.1 Current Classification Based on Codes

The current strategy classifies columns based on codes, identified by the combination of:

- commercial characteristics, such as size, materials, lapping, and thickness;
- technical specifications, such as quality, shade, caliber, and production batch.

Each column can hold only one single code. In case all columns have already been destined to different codes, the system can store distinct codes in the same column, causing penalties.

2.4.2 Proposed Classification Based on Destinations

This study focuses on the definition of a proposed storage policy, categorizing pallets according to their final destinations rather than their codes. Pallets with different downstream destinations, established by in-house logistics flows, are moved to dedicated company warehouses. Therefore, each destination may require pallets to be classified based on further characteristics. The following list outlines the range of possible destinations and indicates the corresponding classes for each one:

- *inter-company pallets*, containing tiles produced for other companies of the same holding. In this case, pallets are classified based on the commercial characteristics of the products;
- *under-choice pallets*, containing non-first quality tiles due to LHP. Such pallets are commercialized in a different commercial channel and, therefore, are shipped to a specific facility. In this case, pallets are not classified based on further characteristics;
- *picking pallets*, destined for picking operations, in which loading units containing fewer tiles than regular-size pallets are prepared to meet sales requirements. In this case, pallets are not classified based on further characteristics;
- *high-rotation pallets*, containing frequently sold goods. They are stored in a warehouse closer to the shipping area. In this case, pallets are sold, and consequently classified, based on their code, that comprehend both technical and commercial characteristics;
- *low-rotation pallets*, containing goods that are sold less frequently. They are stored in a warehouse farther from the shipping area. Also in this case, pallets are sold, and consequently classified, on the basis of their code.

Each column of the buffer area is dedicated to a single class. As in the current storage policy, if all columns have already been destined to other classes, the new policy allows mixing pallets to avoid any stop in the production flow. In this case, whenever a new pallet has to be stored in an already classified column, there are two possible scenarios:

- *soft-mix*: the pallet is stored inside a column dedicated to the same destination, but to a different class. For example, a *soft-mix* class column can contain different codes of inter-company pallets;
- *mix*: the pallet is stored inside a column dedicated to a totally different destination. For example, a *mix* class column can contain both under-choice and picking pallets.

The new classification storage policy was rigorously analyzed through a DES to quantify its impact on major performance indicators.

2.5 Simulation

In this section, we provide a description of the simulation study, including the input data provided to model the process, the conceptual model used to reproduce the real-world system, the verification and validation processes, and the performance indicators selected to measure the effectiveness of the proposed solution.

2.5.1 Input Data

The simulation is based on real world data that reproduces the manufacturing jobs of 30 different days in the company. Therefore, the initiation of the simulation requires an input file for each simulated day, containing real information on each pallet that arrived in the buffer area. The relevant data contained for each pallet includes the identification number and the size of the pallet, its arrival time inside the buffer area, the production characteristics of the tiles, and the information about its downstream destination, such as the level of rotation and whether the pallet is inter-company, under-choice or picking. To generate the necessary input data about downstream destination, a pre-processing step was executed on historical data, calculating the information using company sales data. Since LHP particularly affect the quality of the ceramic tiles, special attention was dedicated to under-choice pallets, to consider their stochastic fluctuations. To gain insights into the variability of under-choice pallet, a histogram was initially plotted to visualize the distribution of the number of pallets. Subsequently, the Shapiro-Wilk statistical test was performed, confirming that the number of daily under-choice pallets follows a normal distribution ($mean = 28.89, std = 8.26$). As a result, we used this distribution to stochastically generate data and we performed 10 simulation runs per day.

2.5.2 Conceptual Model

The simulation aims to improve the utilization of the buffer area by determining the best storage classification policy. The flow of the simulation is schematized in Figure 2.2. At

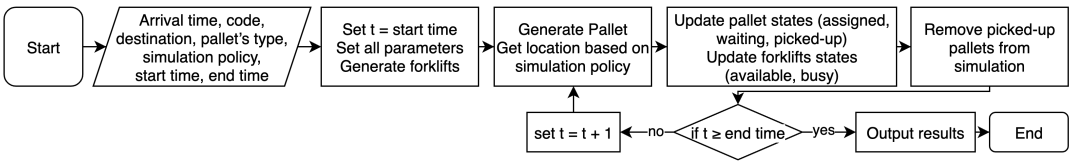


FIGURE 2.2: Simulation flow.

first, the simulation is initialized with the necessary input parameters. Each pallet undergoes four key phases:

- *Creation and Classification*: after being generated, the pallet is classified according to either the characteristics-based or the destination-based policies;
- *Storage*: the pallet's storage location in the buffer area is determined based on space availability, with the westernmost position being favored. If no space is available, the pallet is removed from the simulation and considered lost, according to the real-world process, in which if there is no space available then the production line must stop. Automated Guided Vehicle (AGVs) transport pallets to the storage area. The transportation time follows a triangular distribution derived from real-world data with $min = 90s$, $max = 180s$, and $mode = 120s$.
- *Queuing*: the pallet waits at the storage location;
- *Pick-Up*: the pallet is removed from the buffer area by forklift operators and exits the simulation. Two forklift operators are responsible for the emptying process during working hours and they are considered available when there are not pallets waiting to be removed.

The simulation model was developed by extending Salabim's built-in classes and regulating its flow through the construction of built-in component routines. The main simulation components extended from Salabim's built-in classes are *PalletClass*, *PalletGeneratorClass*, and *ForkliftClass*. *Operators*, *BufferArea*, *ColumnState*, and *Animation* are entities created respectively from Salabim's built-in classes Resources, Queue, State, and Animate.

Figure 2.3 illustrates the logic of the conceptual model through the pseudocode. The

PalletGeneratorClass

```
 1: for pallet in input_data do
 2:     wait until arrivaltime = sim.time
 3:     Pallet(pallet)
 4: end for
```

ForkliftClass

```
 5: while True do
 6:     Wait workingtime
 7:     if bufferqueue is not empty then
 8:         pallet = getPalletToRemove()
 9:         Activate pallet
10:     else
11:         Passivate self
12:     end if
13: end while
```

PalletClass

```
14: x, y, z = storePolicy(self)
15: if x ≠ 1 and y ≠ 1 and z ≠ 1 then
16:     self.location = x, y, z
17: else
18:     return noSpace(self)
19: end if
20: Enter bufferqueue
21: Update animation
22: for forklift in forklifts do
23:     if forklift.state = Passive then
24:         Activate forklift
25:     end if
26: end for
27: Passivate self
28: Leave bufferqueue
29: Update animation
```

FIGURE 2.3: Conceptual model's pseudocode.

PalletGeneratorClass generates pallets according to the input data (Line 3), ensuring that the creation of pallets within the simulation corresponds to the actual arrival times specified by the input data. After the generation of each pallet, the associated routine of *PalletClass* (Line 14) is automatically executed. The routine determines the pallet's location within the buffer area, the placement of the pallet in the appropriate queue, the activation of the *ForkliftClass* if it is in a passive state (Line 24), and finally the passivation of the pallet (Line 27). The *ForkliftClass* models the activities of the operators by removing pallets from the buffer area during working hours, emptying one column at a time. When the pallet is reactivated by *ForkliftClass* (Line 9), it exits the buffer area queue, updating the 2D animation representation.

2.5.3 Verification and Validation

The conceptual model was verified to confirm its accuracy. To ensure verification, the documentation was kept up-to-date throughout the model construction process and the accuracy of the model was evaluated by analyzing the output while modifying the input parameters. Moreover, Salabim's 2D animation was used to visually inspect the model's logic. Figure 2.4 represents a temporal snapshot of the simulation, in which different colors are used to represent distinct classes, in order to highlight any presence of mixed columns. For each cell we also report the number of stacked pallets. When a cell contains four pallets, it is considered to have reached its maximum capacity, and any additional pallets belonging to the same class are assigned to the next cell within the same column. Once a column is completely filled, a new column is then used. Cells with a value equal to zero represent positions that are not occupied by pallets.

Finally, validation was achieved by comparing the average waiting time of pallets in the buffer area for simulation results and actual data regarding 30 different days. Since waiting times were normally distributed, the paired t-test was selected to compare results and determine a statistical difference. The test did not show any significant difference in the average

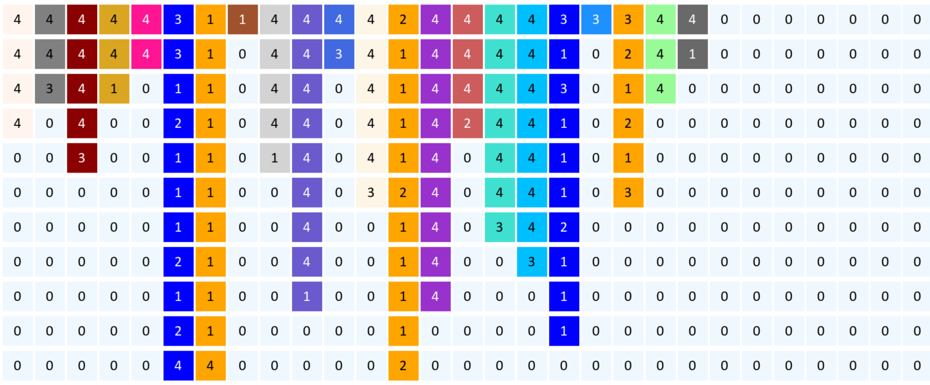


FIGURE 2.4: Simulation 2D animation.

time of the 30-day data (p-*value* > 0.9) and, as a result, the simulation is considered accurate and a valid representation of the real-world process.

2.5.4 Performance Indicators

The purpose of the simulation was to assess possible improvements in the buffer area in terms of space utilization and ease of operation. The simulation assessed the performance of a company in which the production process never stops, while the logistics department operates on a two-shift schedule, as common in the ceramic sector. As a result, during night shifts and weekends, pallets are stored in the buffer area without being retrieved. Therefore, two distinct sets of indicators were collected to evaluate both the described scenarios – with and without operators. In the scenario without operators, the following performance indicators were assessed to evaluate the impact of the classification policies:

- time (in minutes) elapsed before the first occurrence of the *mix* state in the current policy and the *soft-mix* state in the proposed one;
- time (in minutes) elapsed before the area being filled;
- number of pallets lost due to lack of space.

The real-world system was designed to enable operators to handle the production flow without exceeding the available buffer storage area. Therefore, in the scenario with operators, the focused moved from the probability of the area being filled to the assessment of space utilization, considering the following performance indicator:

- maximum number of columns used at the same time, calculated to assess the feasibility of the classification policy in terms of space utilization.

The performance indicators were collected considering the two classification policies presented in Section 2.4 and the results are compared in Section 2.6.

2.6 Experimental Results

Two scenarios were simulated to compare the current classification policy and the proposed one based on the performance indicators defined in Section 2.5.4. The mean results over 10 runs for each of the 30 simulated working days, together with the corresponding average values (Avg.) for each policy, are presented in Table 2.1. Given that each simulation day begins at the end of the logistics shift and considering that the real-world system was designed

TABLE 2.1: Simulation results.

	Without operators						With operators	
	Minutes before *mix/soft-mix* state		Minutes before area filled		Number of lost pallets		Max. number of columns	
Day	Current	Proposed	Current	Proposed	Current	Proposed	Current	Proposed
1	634.2	774.1	1322.8	1334.7	57.7	55.1	26.0	23.5
2	660.7	779.6	1297.2	1469.6	45.9	0.0	28.5	24.2
3	388.3	486.0	1229.0	1177.8	80.4	108.0	29.0	29.0
4	511.4	695.3	1174.4	1230.9	113.2	68.9	28.3	25.1
5	790.9	1019.2	1391.7	1432.2	43.6	3.2	22.2	16.2
6	453.0	681.7	1280.3	1319.1	83.6	53.9	28.8	25.9
7	776.1	966.9	1421.7	1495.9	0.0	0.0	21.3	17.1
8	527.0	729.4	1395.9	1367.5	14.3	20.7	26.9	23.7
9	602.3	851.0	1382.6	1409.9	32.9	15.8	26.3	22.0
10	881.9	948.7	1474.1	1429.8	0.8	2.5	22.3	18.6
11	600.2	718.9	1311.6	1364.3	49.6	31.5	26.8	23.4
12	759.0	847.9	1467.7	1467.6	0.0	0.0	23.6	20.7
13	715.1	972.9	1305.4	1378.5	49.8	22.6	23.7	19.1
14	618.1	887.4	1309.7	1451.8	61.6	1.5	26.8	21.8
15	764.1	803.0	1370.1	1402.7	16.0	4.8	26.2	21.7
16	481.8	694.8	1356.2	1424.7	55.8	14.7	28.8	24.6
17	747.1	930.6	1361.7	1442.0	34.0	1.0	24.7	21.1
18	601.3	782.8	1327.7	1256.4	0.0	0.0	18.8	17.3
19	530.5	913.7	1262.4	1435.5	85.5	2.7	28.3	23.1
20	1004.8	1099.0	1524.0	1498.4	0.0	2.5	22.2	18.7
21	539.5	724.8	1203.5	1245.6	79.5	70.7	28.1	22.9
22	580.0	758.4	1454.7	1419.5	2.7	4.0	26.2	22.4
23	430.6	534.5	1109.4	1118.6	195.5	203.8	29.0	28.6
24	699.6	910.9	1367.9	1476.5	22.6	1.0	25.6	22.1
25	620.1	836.1	1361.3	1477.7	17.7	0.0	26.4	23.6
26	780.6	1178.9	1488.3	1595.0	3.2	0.0	23.1	19.8
27	646.8	835.1	1368.1	1392.4	43.0	37.5	25.6	20.9
28	734.3	937.4	1525.7	1408.2	0.0	9.5	22.2	18.1
29	878.3	928.0	1436.8	1399.6	3.4	35.5	23.2	20.6
30	385.1	511.5	1157.3	1169.2	150.6	157.9	29.0	28.2
Avg.	644.8	824.6	1348.0	1383.1	44.8	31.0	25.6	22.1

to ensure operators empty the area before the shift ends, the buffer area starts the simulation without any pallets, making warm-up time unnecessary. A statistical comparative analysis was performed on the results to determine a significance difference between the means of the two simulated policies. For normally distributed results, the one tail paired t-test was selected, while non-normally distributed data were compared through Wilcoxon signed-rank test.

Regarding shifts without operators, the test revealed that the mean time elapsed before the first occurrence of the *mix* state in the current classification policy and the *soft-mix* state in the devised policy significantly differ ($p\text{-}value < 10^{-12}$). On average, the *soft-mix* state in the new policy occurs 27.9% minutes later than the *mix* state in the current policy, significantly reducing the operation time. Furthermore, the test was conducted on the mean time elapsed before the area was filled. The test demonstrated a significant difference ($p\text{-}value < 10^{-2}$), with the new policy delaying the area filling time by 2.6% compared to the current one, leading to capacity improvements and consequent reductions in production downtime. Additionally, statistical analysis revealed a significant reduction in the mean number of lost pallets ($p\text{-}value < 10^{-2}$), with an average decrease of 30.8%. The discussed results showed how the proposed policy enhances the flexibility of the storage activities in the scenario without operators. Therefore, the company can reconsider the number of required operators and the distribution of the shifts, resulting in economic benefits. In shifts with operators (reported in the right part of Table 2.1) the test demonstrated a significant difference in the mean number of columns required (*p-value* $< 10^{-13}$), highlighting the possibility to compress the size of the buffer area about the 13.5% without reducing the production capacity. Overall, the experimental results confirmed the benefits of the devised policy, both in terms of economic efficiency and flexibility. Further economical studies were conducted to estimate the potential cost savings associated to the area emptying process, revealing that the proposed policy is expected to reduce the costs by 17%.

The results outlined how the developed policy led to a statistically significant improvement in the performance of the buffer storage area. However, it was essential for the company to comprehend how the new policy would impact different scenarios. This understanding became particularly relevant due to the positive market trends observed in the ceramic sector and particularly within the company. Therefore, the assumption of a percentage increase in production was made to assess the potential impact of the policy under varying conditions. Specifically, seven different scenarios were simulated, incrementally increasing production quantity by 5%, from 0% to 30%. For each scenario, the simulation was conducted over a period of 30 days, and the mean results of 10 runs per scenario are reported in Table 2.2. To

TABLE 2.2: Average simulation results for increasing percentages of production quantity.

	Without operators						With operators	
	Minutes before mix/soft-mix state		Minutes before area filled		Number of lost pallets		Max. number of columns	
Increase in production (%)	Current	Proposed	Current	Proposed	Current	Proposed	Current	Proposed
0	644.8	824.6	1348.0	1383.1	44.8	31.0	25.6	22.1
5	599.0	767.2	1269.8	1295.8	77.0	62.6	26.6	23.7
10	580.4	735.5	1203.2	1232.3	112.9	98.7	27.3	24.7
15	552.8	701.9	1148.2	1176.9	151.7	137.3	28.3	25.9
20	532.8	672.9	1098.7	1123.9	190.9	177.5	28.4	26.7
25	506.3	643.7	1050.9	1078.6	232.7	216.7	28.6	27.4
30	487.1	617.3	1012.2	1038.6	269.7	256.9	28.8	27.8

assess the efficiency of the proposed policy, statistical tests were conducted, which revealed that the devised policy outperformed the current policy in all seven scenarios, regardless of the production increase. Figure 2.5 presents four graphs, each one displaying the mean simulation results for a distinct performance indicator. The first graph shows a significant increase in the time elapsed before the occurrence of the *mix/soft-mix* state with the proposed policy for all production quantities, resulting in a steady improvement in performance of approximately 27.2% for each scenario. Furthermore, the second graph demonstrates that the time before the area is filled improves for all production levels, saving approximately 2.4% of the time. The third graph shows that the number of lost pallets decreases with the new policy, although the percentage improvement decreases with an increase in production, due to an increase in the absolute number of lost pallets. Moreover, the number of storage columns used during operator shifts is shown in the fourth graph and it decreases by an average of 7.9%, allowing for the possibility to reduce the buffer area size. Overall, the proposed policy demonstrates its effectiveness in all scenarios, highlighting its potential to improve performance and enhance economic and flexibility improvements.

2.7 Conclusion

The study aimed at determining the best strategy to temporarily store products within a buffer area in an Italian ceramic tile company, in a highly uncertain context characterized by LHP.

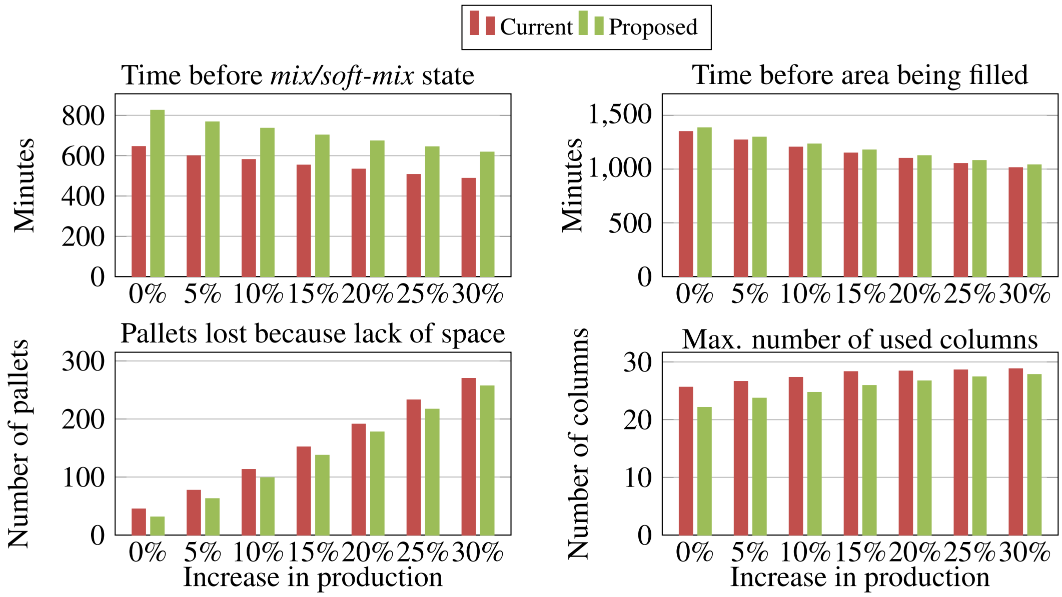


FIGURE 2.5: Mean simulation results for increasing percentages of production quantity.

The identification of an effective policy is crucial for ensuring streamlined material flow and reducing operational costs. In particular, in a highly variable production scenario, where homogeneous batches are essential to meet customer demand, an effective storage policy must classify products into homogeneous subgroups. The policy proposed by this study focused on the subdivision of products in the buffer area according to their downstream destination, rather than their technical production characteristics, as it is currently done by the company. The approach aimed at improving the buffer area management, increasing available space and reducing the intermixing of products belonging to different sub-categories. To compare the effectiveness of the current and proposed policies, a discrete event simulation was conducted using Salabim. The model was rigorously verified and validated, and simulation results were obtained for both policies and compared.

The results indicated a statistical difference between the two policies. In particular, it was demonstrated that the new policy improved the column classification of the buffer area, consequently reducing the number of required columns. This provided the company with the opportunity to either reduce the buffer area's space or increase production capacity. Also, the proposed policy delayed the filling of the area, preventing production stoppages that could negatively and significantly impact the company's profits. In addition, the new policy reduced the number of times in which non-homogeneous subgroups were in the same column, consequently accelerating operators' tasks. Furthermore, besides the measured results, pallet subdivisions based on downstream classification could further reduce the time required for subsequent operations. Therefore, the company could profit from both economic and flexibility advantages, providing strategic benefits to managerial decision-makers. Moreover, the sensitivity analysis showed that the new policy benefits the performance even in the scenario of increased production quantity. For these reasons, the company decided to change their storage policy according to the results of the study, classifying their products in the buffer area based on downstream destination.

Finally, the simulation model developed in this study is a valid representation of the process and can be utilized for further research regarding new classification policies for continuous improvement.

References

Abdolazimi, Omid et al. (2021). “Designing a New Mathematical Model Based on ABC Analysis for Inventory Control Problem: A Real Case Study”. In: *RAIRO-Operations Research* 55.4, pp. 2309–2335.

ACIMAC Research Department (Oct. 2022). *World Production and Consumption of Ceramic Tiles*. Tech. rep. 10. Manufacturing Economics Study.

Alemany, MME, A Ortiz, and Vicente S Fuertes-Miquel (2018). “A Decision Support Tool for the Order Promising Process With Product Homogeneity Requirements in Hybrid Make-to-Stock and Make-to-Order Environments. Application to a Ceramic Tile Company”. In: *Computers & Industrial Engineering* 122, pp. 219–234.

Andreola, Fernanda et al. (2016). “Recycling of Industrial Wastes in Ceramic Manufacturing: State of Art and Glass Case Studies”. In: *Ceramics International* 42.12, pp. 13333–13338.

Anglano, Cosimo, Massimo Canonico, and Marco Guazzone (2019a). “Online User-Driven Task Scheduling for FemtoClouds”. In: *Fourth International Conference on Fog and Mobile Edge Computing*. Institute of Electrical and Electronics Engineers, Inc. Piscataway, NJ, pp. 5–12.

Anglano, Cosimo, Massimo Canonico, and Marco Guazzone (2019b). “WQR-UD: an Online Scheduling Algorithm for FemtoClouds”. In: *Proceedings of the 12th EAI International Conference on Performance Evaluation Methodologies and Tools*. Ed. by Franceschinis Giuliana A., Prabhu Balakrishna J., and Touati Mikaël. Association for Computing Machinery. New York, NY, pp. 179–182.

Atılgan Türkmen, Burçin, Tuba Budak Duhbacı, and Şeyma Karahan Özbilen (2021). “Environmental Impact Assessment of Ceramic Tile Manufacturing: A Case Study in Turkey”. In: *Clean Technologies and Environmental Policy* 23, pp. 1295–1310.

Baldwa, Vaibhav et al. (2020). “A Combined Simulation and Machine Learning Approach for Real-Time Delay Prediction for Waitlisted Neurosurgery Candidates”. In: *Proceedings of the 2020 Winter Simulation Conference*. Ed. by Ki-Hwan G. Bae et al. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc., pp. 956–967.

Baraldi, Luca (2022). “World Production and Consumption of Ceramic Tiles”. In: *Ceramic World Review* 148, pp. 36–57.

Boza, Andrés et al. (2014). “A Model-Driven DSS Architecture for Delivery Management in Collaborative Supply Chains With Lack of Homogeneity in Products”. In: *Production Planning & Control* 25.8, pp. 650–661.

Dahl, Ole-Johan and Kristen Nygaard (1966). “SIMULA: An ALGOL-based simulation language”. In: *Communications of the ACM* 9.9, pp. 671–678.

Davoli, Giovanni, Sergio Amedeo Gallo, and Riccardo Melloni (2008). “Analysis of Industrial Processes Based on Integration of Different Simulation Tools”. In: *Proceedings of the 10th International Conference of the Modern Information Technology in the Innovation Processes of the Industrial Enterprises*. Ed. by Hán Jan and Holejšovská Pavla. Prague, Czech Republic, 38–49.

Davoli, Giovanni et al. (2010). “A Stochastic Simulation Approach for Production Scheduling and Investment Planning in the Tile Industry”. In: *International Journal of Engineering, Science and Technology* 2.9, pp. 107–124.

Dondi, Michele et al. (2021). “Resource Efficiency Versus Market Trends in the Ceramic Tile Industry: Effect on the Supply Chain in Italy and Spain”. In: *Resources, Conservation and Recycling* 168.5, p. 105271.

Dubey, Avanish Kumar and Vinod Yadava (2008). “Laser Beam Machining – A Review”. In: *International Journal of Machine Tools and Manufacture* 48.6, pp. 609–628.

Efthymiou, Orestis K and Stavros T Ponis (2019). “Current Status of Industry 4.0 in Material Handling Automation and In-House Logistics”. In: *International Journal of Industrial and Manufacturing Engineering* 13.10, pp. 1368–1372.

Erden, Caner, Halil Ibrahim Demir, and Onur Canpolat (2021). “A Modified Integer and Categorical PSO Algorithm for Solving Integrated Process Planning, Dynamic Scheduling and Due Date Assignment Problem”. In: *Scientia Iranica*. Forthcoming.

Erden, Caner, Halil Ibrahim Demir, and Abdullah Hulusi Kökçam (2019). “Solving Integrated Process Planning, Dynamic Scheduling, and Due Date Assignment Using Metaheuristic Algorithms”. In: *Mathematical Problems in Engineering* 2019, p. 1572614.

Ferrari, Anna Maria et al. (2019). “Building a Sustainability Benchmarking Framework of Ceramic Tiles Based on Life Cycle Sustainability Assessment (LCSA)”. In: *Resources* 8.1, p. 11.

Ham, Ruud van der (2018). “Salabim: Discrete Event Simulation and Animation in Python”. In: *Journal of Open Source Software* 3.27, pp. 767–768.

Lang, Sebastian et al. (2020a). “Evolving Neural Networks to Solve a Two-Stage Hybrid Flow Shop Scheduling Problem with Family Setup Times”. In: *Proceedings of the 2020 Hawaii International Conference on System Sciences*. Ed. by Tung X. Bui. University of Hawaii at Manoa, pp. 1298–1307.

Lang, Sebastian et al. (2020b). “Integration of Deep Reinforcement Learning and Discrete-Event Simulation for Real-Time Scheduling of a Flexible Job Shop Production”. In: *Proceedings of the 2020 Winter Simulation Conference*. Ed. by Ki-Hwan G. Bae et al. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc., pp. 3057–3068.

Lang, Sebastian et al. (2021a). “Modeling Production Scheduling Problems as Reinforcement Learning Environments Based on Discrete-Event Simulation and OpenAI Gym”. In: *IFAC Papers OnLine* 54.1, pp. 793–798.

Lang, Sebastian et al. (2021b). “NeuroEvolution of Augmenting Topologies for Solving a Two-Stage Hybrid Flow Shop Scheduling Problem: A Comparison of Different Solution Strategies”. In: *Expert Systems with Applications* 172, p. 114666.

Lang, Sebastian et al. (2021c). “Open-Source Discrete-Event Simulation Software for Applications in Production and Logistics: An Alternative to Commercial Tools?” In: *Procedia Computer Science* 180, pp. 978–987.

Ma, Shuaiyin et al. (2022). “Digital Twin and Big Data-Driven Sustainable Smart Manufacturing Based on Information Management Systems for Energy-Intensive Industries”. In: *Applied Energy* 326, p. 119986.

Mangi, Sajjad Ali et al. (2022). “Recycling of Ceramic Tiles Waste and Marble Waste in Sustainable Production of Concrete: A Review”. In: *Environmental Science and Pollution Research* 29.13, pp. 18311–18332.

Medina-Salgado, María Sonia et al. (2021). “Adaptive Life Cycle Costing (LCC) Modeling and Applying to Italy Ceramic Tile Manufacturing Sector: Its Implication of Open Innovation”. In: *Journal of Open Innovation: Technology, Market, and Complexity* 7.1, p. 101.

Nadir, Uroosa et al. (2020). “Performance Improvement Using Simulation Tool in a Tiles Production Facility”. In: *Mehran University Research Journal Of Engineering & Technology* 39.1, pp. 117–132.

Sciancalepore, Corrado, Tiziano Manfredini, and Federica Bondioli (2014). “Antibacterial and Self-Cleaning Coatings for Silicate Ceramics: A Review”. In: *Advances in Science and Technology* 92, pp. 90–99.

Shoaib, Mohd and Varun Ramamohan (2022). “Simulation Modeling and Analysis of Primary Health Center Operations”. In: *Simulation* 98.3, pp. 183–208.

Soares, Armindo, Carina Pimentel, and Ana Moura (2022). "Production Planning and Scheduling With Applications in the Tile Industry". In: *International Journal of Operational Research* 44.1, pp. 58–79.

Xie, Xianghua (2008). "A Review of Recent Advances in Surface Defect Detection Using Texture Analysis Techniques". In: *Electronic Letters on Computer Vision and Image Analysis* 7.3, pp. 1–22.

Zanelli, Chiara et al. (2021). "Waste Recycling in Ceramic Tiles: A Technological Outlook". In: *Resources, Conservation and Recycling* 168, p. 105289.

Zhao, Zhike (2021). "Review of Non-destructive Testing Methods for Defect Detection of Ceramics". In: *Ceramics International* 47.4, pp. 4389–4397.

Chapter 3

Decision Support Systems in Ceramic Tile Industry [1]

In this this study we present an automated and optimized Decision Support System (DSS) that includes an integer linear programming (ILP) model. The DSS optimizes the order management process by determining optimal load configurations from peripheral warehouses onto transport vehicles. The resulting transportation plan, generated through this approach, aims to meet customer demands while minimizing overall costs. Computational tests, conducted on a real-world case study, validated the efficiency of the proposed system.

3.1 Introduction

One of the critical challenges faced by industries is intralogistics, the logistics component that take place within the company. Intralogistics involves two main functions: internal transport of materials, and information flow management. The former includes the movement of products between different production plants and warehouses, while the latter refers to software systems that tracks the movements of the physical goods. Both functions are essential to ensure logistics efficiency and must be effectively integrated. In addition, in a large number of companies products are handled between different warehouses, consuming valuable space and operational time. Therefore, warehouse management plays a crucial role in ensuring efficiency and reducing logistics expenses.

This study focuses on order management in a business context characterized by a central shipping site for orders consolidation and various peripheral storage sites for production and stocking. Similar examples can be found in the literature related to the transshipment problem (Chiou, 2008), where models are used to decide how to move stocks between warehouses of the same company to satisfy the demand (Patil et al., 2021). Particular attention to this topic is given in the online retailing context, in which individual stock units are shipped to central warehouses to consolidate orders (Zhang et al., 2021). Some authors also incorporate the selection of transportation modes in the model (Mishra et al., 2023). Moreover, studies have explored the profitability of integrating package selection into the shipping decisions by integrating different unit configurations (Li et al., 2020).

Despite the primary focus on product transshipment in this study, the presence of a central shipping point and distributed warehouses makes our problem similar to the supplier selection problem (Chai et al., 2013). In this scenario, the central depot and the distributed warehouses can be viewed as the plant and the individual suppliers, respectively. Many studies address the issue of supplier selection and order quantity allocation in multi-stage supply

[1] The results of this chapter appears in : Dotti, G., Iori, M., Subramanian, A. and Taccini, M. "An Integrated Decision Support System for Intra-logistics Management with Peripheral Storage and Centralized Distribution". In *Proceeding of the International Conference on Enterprise Information Systems, 2024, pp. 612-619.*

chain (Pazhani et al., 2016), more precisely, for those companies with many potential suppliers (Mendoza and Ventura, 2008). Some authors also consider how to assign shipment to different modes of transportation (Glickman and White, 2008). However, to the best of our knowledge no study integrates simultaneous decisions about warehouses, optional feature, stock unit configuration, and transportation modes.

Also, in many companies, order management is manual, involving stages that slow down the process and consume resources in non-value added activities. Human decisions in the process may lead to errors or sub-optimal outcomes. Hence, a Decision Support System (DSS) for supplier selection was proposed in the literature (Scott et al., 2015), as well as for order allocation problems (Erdem and Göçen, 2012).

This study proposes an automated and optimized DSS to enhance order management in production companies. Automation is achieved by fully integrating the proposed software architecture into the company's existing procedures, thereby eliminating inefficiencies associated with non-value-added activities. Optimized decisions are achieved by means of an integer linear programming (ILP) model, which selects goods from peripheral warehouses and arranges loads on transport vehicles, reducing inefficiencies related to human decisions and minimizing total order management costs. The research is inspired by a real-world case study arising in the ceramic tile production (discussed in detail in Section 3.5 below), but it is very general and can encompass a variety of applications.

The remainder of this chapter is structured as follows. Section 3.2 presents a complete problem description. Section 3.3 focuses on detailing the decision support system. Section 3.4 outlines the mathematical model used for optimization. Section 3.5 presents the real-world case study and Section 3.6 discusses the results obtained. Lastly, Section 3.7 summarizes the study and presents future research directions.

3.2 Problem Description

This section provides a comprehensive overview of the problem by exploring both functions of intralogistics. It delves into materials flow in Section 3.2.1 and information flow in Section 3.2.2.

3.2.1 Material Flow

The primary challenge is efficiently fulfilling incoming orders, requiring goods transportation from peripheral warehouses to a central facility for order consolidation and customer shipment.

Each order requests a single item along with a specified number of boxes. Multi-line orders can be simplified by preprocessing and segmenting them into separate orders, each with a single order line. Orders may also specify additional product features. In this context, a feature refers to a distinguishable attribute or characteristic of the products, such as their color or shade, that the client can specify when placing an order. If the feature is specified by the client, the preference must be respected throughout the order fulfillment. On the other hand, when a client does not explicitly request a specific feature for the order, the company has the flexibility to select it. Nevertheless, in both scenarios, it is essential to ensure that all boxes shipped for the same order have not only the same item, but also the same chosen feature to ensure order homogeneity.

Furthermore, each item and feature may have various pallet configurations, each containing a specific number of boxes. It should be noted that pallets cannot be divided into smaller units.

Items are stored in various warehouses, each with different travel times from the central depot and stocked with specific pallet configurations for items with certain features. Picking each box incurs a cost depending on the warehouse. In addition, peripheral warehouses can be accessed via different transportation options, each with an hourly cost and weight capacity. Each box contains copies of a single item, with its weight depending on the item's weight. The set of boxes loaded onto a mode of transport must adhere to its capacity, and each mode can only serve one warehouse per transfer order release.

The optimization process involves several decisions: (i) assigning a feature to orders without specifications; (ii) determining the number of pallets of each configuration to pick from each warehouse; (iii) allocating each mode of transportation to a single warehouse; and (iv) designing how to load the picked pallets onto modes of transportation to respect the capacity. In some companies, the decision-making process is entirely manual, with an operator deciding based on their judgment. This study aims to meet demand while minimizing total transport and retrieval costs and enhancing system performance.

3.2.2 Information Flow

The material flow outlined in Section 3.2.1 requires a cohesive information flow to track operations and order status. Typically, the information flow involves manual steps carried out by various stakeholders:

- sales representatives initiate the process by emailing logistics operators for goods transportation;
- logistics operators aggregate requests, waiting until they have enough to fill at least one transfer capacity. Once the threshold is reached, they manually organize transportation logistics, making decisions based on their expertise;
- decisions are communicated via email to the commercial department;
- upon items reaching the centralized distribution center, the logistics department manually notify sales representatives;
- sales representatives input the newly arrived item into the order management software to progress order fulfillment.

The described process is costly, resulting in slow and repetitive operations that consume valuable time and resources and ultimately provide little added value to the end customer. Some of the most prevalent issues include:

- the fulfillment of each order requires numerous manual steps, resulting in time inefficiencies;
- since each sales representative initiates an independent information flow, visibility on available items is compromised. This lack of awareness among sales representatives may lead to the same pallet in stock being requested for two distinct orders, as representatives are unaware of each other's requests;
- as previously indicated, the picking process exclusively accommodates orders for complete pallets. Consequently, order quantities must be rounded up. In a situation where two operators require the same product in quantities less than a full pallet, they may have the option to combine their orders, approximating to one pallet instead of two. However, the lack of mutual awareness among operators about each other's orders precludes the effective aggregation of quantities, resulting in the costly picking of unnecessary products;

- the process is highly dependent on both total loads and operators availability, making it inherently non-scalable;
- as a significant amount of time elapses from the initial request, the sales department may repetitively solicit the logistics team via email, placing an additional workload on the operators.

In response to the identified challenges, this study aims to automate and digitalize the process, with the goal of reducing logistic operator overhead, improving response time, and improving process scalability.

3.3 Digitalization

3.3.1 Process Overview

As outlined in Section 3.2.2, the digitalization of the information flow is designed to reduce the workload overhead for both sales representatives and logistics operators. To address this issue, we developed a DSS, which is extensively described in this section.

The new digitalized flow follows four main steps. The first step, schematized in Figure 3.1, is executed periodically and involves reading orders from the Enterprise Resource Planning (ERP) system to populate the database. Such orders contain the required information

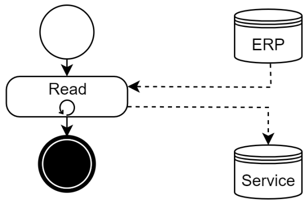


FIGURE 3.1: Reading component of DSS architecture.

and are manually added to the ERP system by sales representatives.

The second, third, and fourth steps, schematized in Figure 3.2, are executed consecutively when the optimization time is reached. The second step performs a check to ensure

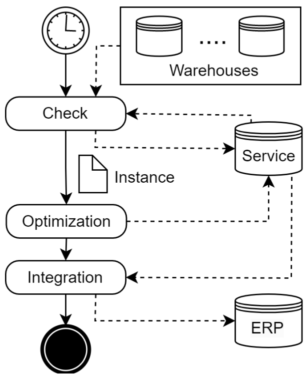


FIGURE 3.2: Control, optimization, and integration components of DSS architecture.

that there is enough stock in the peripheral warehouses to fulfill all orders. If inadequacies are identified, unsatisfiable orders are flagged in the service database and excluded from subsequent steps. Moreover, notifications are dispatched to the respective sales representatives who added these unsatisfiable orders. On the contrary, if the orders are satisfiable, the software generates an instance for the optimization step.

Subsequently, the third step involves the execution of the optimization model, described in Section 3.4. Upon completion of the optimization step, the database is updated with new decisions, such as selected features, chosen warehouses, pallet types, and transportation configurations.

Finally, the fourth step is integrated into the ERP system. Specifically, this step considers all decisions made by the optimization step from the service database and, through APIs, transmits them to the ERP system. Once this integration is completed, the operators of the peripheral warehouses gain visibility into the items they need to prepare for transportation.

3.3.2 Technologies

We have designed a user-friendly DSS to help companies manage customer orders efficiently and make optimization algorithms accessible to non-experts. Our DSS runs on Docker, a software platform for developing and deploying applications in isolated containers. Docker ensures future scalability, portability and accelerates deployment, keeping the hosting machine unmodified.

In particular, three distinct containers have been developed:

- *Service container*, that hosts the MySql service database for storing transfer requests and monitoring their progression through different stages.
- *Job orchestrator container*, equipped with database connection drivers, Python and Node-RED. It includes different Python-based jobs used to build the new digitalized flow that is scheduled directly by Node-RED. If necessary, this flow can also be run manually to ensure flexibility. This flexibility proves beneficial, especially when the daily volume of requests is low, allowing the logistics department to defer optimization until subsequent days to accumulate more orders and efficiently plan material transportation.
- *User interface container*, to assist sales representatives in monitoring their orders we developed an intuitive and user-friendly web interface. The interface is built on Flask micro-web framework, written in Python, which encompasses both a back-end and a front-end.

3.4 Optimization

This section provides a formal definition of the optimization problem addressed in this work, as well as an ILP-based mathematical model.

3.4.1 Problem Definition

The problem we face can be formalized as follows. A set I includes different items, each characterized by its weight w_i. The set J represents orders, each requiring one item i in quantity d_{ij}. Additionally, orders may specify a desired feature from the features set K: if feature k is chosen for order j, the corresponding parameter f_{jk} is set to 1; otherwise, it is set to 0.

Set H represents peripheral warehouses, each defined by the travel time r_h from the central depot and the processing cost u_h per box. Items can be arranged in different pallet configurations, contained in set P, each counting q_{ip} boxes. Pallet configurations cannot be split into smaller units. Each warehouse h maintains a stock s_{hpik} of item i with feature k in pallet configuration p.

Finally, set T denotes the modes of transport, each characterized by a capacity c_t and an hourly cost m_t.

A feasible solution for the problem must satisfy the following constraints: (i) each feasible order request is fulfilled, providing items with uniform features; (ii) if specified, the feature must respect customers' choice; (iii) pallets of items must be picked from the warehouses according to their stock; (iv) picked pallets must be loaded into modes of transportation according to their capacity; (v) each mode of transportation can perform only one route in a single day, visiting a single warehouse. The objective of the problem is to obtain a feasible solution that minimizes the total cost of order management, including transportation and internal movement costs.

Note that the problem described above generalizes the well-know bin packing problem, which is NP-hard, when we consider a single warehouse ($|H| = 1$), no optional features ($|K| = 0$), no pallet configurations ($|P| = 0$), transports with identical capacities (c_t is constant, $\forall t \in T$), unitary transportation costs ($m_t = \frac{1}{r_h}$, $\forall t \in T$, $\forall h \in H$), and no retrieval costs ($u_h = 0$, $\forall h \in H$). Therefore, our problem is also NP-hard.

3.4.2 Mathematical Formulation

Let x_{jk} be a binary variable that takes the value 1 if feature k is assigned to order j and 0 otherwise. An integer variable y_{hpik} identifies the number of pallets of item i in feature k with pallet configuration p picked from warehouse h. An integer variable z_{hpit} specifies the number of pallets of item i in pallet configuration p loaded onto modes of transport t departing from warehouse h. Lastly, let v_{ht} be a binary variable that is equal to 1 if mode of transportation t is assigned to warehouse h and 0 otherwise. An ILP formulation for the problem can be expressed as:

$$\min \quad \sum_{h\in H}\sum_{t\in T} m_t r_h v_{ht} + \sum_{h\in H}\sum_{p\in P}\sum_{i\in I}\sum_{k\in K} u_h q_{ip} y_{hpik} \tag{3.1}$$

$$\sum_{k\in K} x_{jk} = 1, \qquad j \in J \tag{3.2}$$

$$x_{jk} \geq 1, \qquad j \in J, k \in K : f_{jk} = 1 \tag{3.3}$$

$$\sum_{h\in H}\sum_{p\in P} q_{ip} y_{hpik} \geq \sum_{j\in J} d_{ij} x_{jk}, \qquad i \in I, k \in K \tag{3.4}$$

$$y_{hpik} \leq s_{hpik}, \qquad i \in I, k \in K, h \in H, p \in P \tag{3.5}$$

$$\sum_{k\in K} y_{hpik} = \sum_{t\in T} z_{hpit}, i \in I, h \in H, p \in P \tag{3.6}$$

$$\sum_{h\in H} v_{ht} \leq 1, \qquad t \in T \tag{3.7}$$

$$\sum_{i\in I}\sum_{p\in P} q_{ip} w_i z_{hpit} \leq c_t v_{ht}, \qquad t \in T, h \in H \tag{3.8}$$

$$x_{jk} \in \{0,1\}, \qquad j \in J, k \in K \tag{3.9}$$

$$v_{ht} \in \{0,1\}, \qquad h \in H, t \in T. \tag{3.10}$$

$$y_{hpik} \geq 0, \text{integer}, \qquad h \in H, p \in P, i \in I, k \in K \tag{3.11}$$

$$z_{hpit} \geq 0, \text{integer}, \qquad h \in H, p \in P, i \in I, t \in T. \tag{3.12}$$

The objective function (3.1) minimizes the transport and picking costs. Constrains (3.2) impose that only one feature is chosen for each order. Constraints (3.3) ensure that the requested feature of a line is respected, keeping consistency with customer requests when indicated. Constrains (3.4) state that the demand of each order is satisfied. Constraints (3.5) prevent the picking of items from warehouses in amounts that exceed their actual stock levels, maintaining the integrity of the inventory. Constraints (3.6) ensure that every picked pallet is shipped. Constraints (3.7) impose that each mode of transport is associated with at most one warehouse, and constraints (3.8) guarantee that the capacity of the modes of transport is not exceeded. Constraints (3.9)–(3.12) describe the domain of the variables.

3.5 Case Study: the ceramic tile industry

The research was conducted in collaboration with an international ceramic tile company headquartered in Italy. Over the last decade, the global tile market has experienced significant growth and increasing importance, with global tile production reaching 16.8 billion square meters worldwide (ACIMAC Research Department, 2023). Focusing on Italy, the country stands out as the leading global exporter by revenue and ranks seventh in production volume. Italy produced 431 million square meters of tiles, generating €7.2 billion in revenue, highlighting the sector's significance for the country. Consequently, intralogistics optimization in the sector is crucial for effectively controlling non-value-added costs.

A notable issue within the ceramic sector is the Lack of Homogeneity in the Product (LHP), a phenomenon arising from uncertain production processes (Alemany et al., 2013). Consequently, despite the utilization of homogeneous inputs, these processes generate heterogeneity in the outputs. This characteristic is particularly relevant in the ceramic industry, due to the use of clays and stochastic elements such as humidity and temperature. Specifically, one of the main tile characteristics affected by LPH is shade, which in this context refers to the variation in color within a particular batch or set of tiles. In industrial manufacturing processes, due to LHP, achieving tiles with the same color shade can be challenging. To address this, manufacturers group tiles based on shade uniformity before packaging to ensure a consistent appearance upon installation. As a result, shade can be addressed as the optional feature outlined in the model: customers have the option to request a specific shade if needed (i.e., to match a previous order). However, even when the shade is not specified, every tile within of the order must be shipped in the same shade to guarantee aesthetic homogeneity.

The ceramic tile company studied has a structure consisting of a central shipping center and two peripheral warehouses. The DSS, outlined in Section 3.3, was implemented and tested using real-world instances collected from the company over a month. It retrieves data from various databases that include information about warehouses, orders, and transportation resources. The aim is to generate an optimized transportation plan that specifies the most efficient load configuration from each warehouse to meet customer demands while minimizing overall costs. This plan is intended to be provided daily or weekly, depending on the number of orders that can be aggregated for efficiency.

3.6 Computational Results

The optimization model was solved using three distinct solvers: Gurobi, CBC, and HiGHS. This approach was chosen to facilitate a comprehensive performance comparison, considering Gurobi's superior performance, as well as the advantageous open-source licenses of HiGHS and CBC. In fact, the company is inclined to purchase the solver license only if the results exhibit significant improvement compared to those provided by the open-source solvers.

Computational experiments were conducted on an Intel(R) Xeon(R) CPU E5-2640 v3 at 2.60 GHz with 64 GB of RAM, running Microsoft Windows 10 and using up to 32 threads. A time limit of 600 s was set, and a relative MIP tolerance of 10^{-4} was imposed.

Table 3.1 presents the computational results for 23 real-world instances, solved with the three different proposed solvers. Since the optimization interval is determined by the company, the number of orders in the instances can be controlled. Therefore, the instances in the table are ordered based on the number of orders, denoted as $|J|$.

For Gurobi we report the objective value expressed as the total cost in euros, the total computing time in seconds, and the time elapsed to find the incumbent solution. Regarding HiGHS and CBC, we provide the gap between the incumbent solution and the lower bound found by the respective solver (Gap_a), as well as the gap between the incumbent solution found by the open-source solvers and the best primal solution found by Gurobi (Gap_b). Specifically, Gap_a and Gap_b are computed as in (3.13) and (3.14), respectively, where i is the incumbent solution value, lb is the lower bound, gs is Gurobi's solution value and glb is Gurobi's lower bound. We also report the computing time and time to achieve the incumbent solution.

$$\text{Gap}_\text{a} = \frac{i - lb}{i} \tag{3.13}$$

$$\text{Gap}_\text{b} = \frac{i - gs}{glb} \tag{3.14}$$

As depicted in the table, Gurobi consistently exhibited rapid convergence to optimality across all instances, except for instances 16 and 23, where it reached the predefined time limit, with a gap of 0.02% for instance 16 and 1.3% for instance 23.

Although open-source solvers HiGHS and CBC may not guarantee optimality within the time limit, a comparison with Gurobi reveals that they often find the optimal solution value. For instance 16, HiGHS and CBC provide identical solutions, whereas HiGHS outperforms CBC on instance 19, 20, 21, and 23. However, in instance 18, CBC achieves the optimal solution value while HiGHS reaches a Gap_b of 0.08%. Comparing HiGHS with CBC, HiGHS reaches the incumbent solution faster for more than half of the instances. Moreover, HiGHS finds the optimal solution in 20 instances, while CBC achieves this in 17 instances, consistently with a better internal gap. Overall, considering the minimal difference between commercial and open-source solvers on the reported instances, exploring the utilization of open-source solvers could lead to potential cost savings for the company.

Given that the process in the case study is carried out manually by operators, the solutions generated by the solver were subsequently compared to the manual calculations performed by operators. Table 3.2 illustrates the comparison between the objective function values computed by the three different solvers and those manually calculated. The table indicates a direct correlation between the total cost of the solution and the instance size, due to the increasing number of required transportations. Consequently, for instances with a small number of orders (e.g., instances 1, 2, and 3), savings are limited as all materials can fit in a single truck, minimizing potential gains. However, as the instance size grows, the manual decision-making complexity also increases proportionally, expanding the possibility of improvement. Therefore, employing an optimization model can lead to cost reductions of up to 40% in material flow. Furthermore, on average, all solvers demonstrate savings of at least 24% compared to the operators' manual solutions. Notably, even for instance 23, which was not optimally solved by any of the solvers, a substantial 28% reduction in costs was achieved.

Moreover, the savings are significantly enhanced by the digitalization of the information flow, leading to a reduction in time allocated to non-value-added activities. To quantify this enhancement, an estimation of the time required by operators for the manual steps described in Section 3.2.2 was conducted within the company. The time required for the operator is heavily dependent on the number of orders received. On average, the company estimated that 40-50 requests are received per day, requiring a logistic operator's commitment of 4 hours. However, it is crucial to note that for increasing workloads, the required time grows more

TABLE 3.1: Computational results of the real-world instances solved with Gurobi, HiGHS and CBC.

Instance		Gurobi			HiGHS				CBC			
#	$\|J\|$	Obj. Value (€)	Total Time (s)	Time Incumbent (s)	Gap_a (%)	Gap_b (%)	Total Time (s)	Time Incumbent (s)	Gap_a (%)	Gap_b (%)	Total Time (s)	Time Incumbent (s)
1	7	295.07	0.08	0.02	0	0	0.18	0.02	0	0	26.14	2.56
2	10	301.14	0.05	0.01	0	0	0.12	0.10	0	0	1.89	0.44
3	30	332.63	0.06	0.03	0	0	0.99	0.60	0	0	85.39	1.39
4	50	352.24	0.08	0.02	0	0	0.54	0.50	0	0	59.45	1.00
5	53	666.75	1.46	1.44	0	0	8.06	3.40	45.35	0	tlim	2.67
6	70	471.40	0.69	0.04	0	0	2.14	1.80	14.53	0	tlim	1.52
7	81	763.16	1.74	0.32	0	0	7.71	3.10	8.16	0	tlim	3.42
8	95	941.47	3.03	2.80	0	0	304.36	12.00	21.78	0	tlim	14.96
9	100	770.34	1.81	0.30	0	0	5.44	0.90	5.50	0	tlim	3.94
10	103	900.52	1.94	0.33	0	0	6.52	2.80	5.22	0	tlim	121.86
11	106	3983.02	18.90	2.10	2.10	0	tlim	194.30	2.21	0.01	tlim	507.87
12	107	1234.02	2.64	0.80	0	0	465.27	5.80	11.04	0	tlim	110.00
13	108	1102.63	2.56	2.44	10.80	0	tlim	39.10	13.06	0	tlim	76.75
14	115	1204.98	2.33	0.94	2.10	0	tlim	10.40	16.23	0	tlim	14.67
15	121	905.31	1.87	0.61	0	0	149.54	8.90	11.00	0	tlim	4.26
16	164	1896.26	tlim	600.85	5.20	4.56	tlim	62.40	5.47	4.56	tlim	122.95
17	167	1557.90	14.22	11.62	6.10	0	tlim	76.10	7.90	0	tlim	108.23
18	179	2001.93	13.90	12.06	6.00	0.08	tlim	65.40	6.28	0	tlim	266.78
19	190	1985.83	40.62	40.59	4.20	0	tlim	62.7	4.56	0.02	tlim	10.08
20	201	2083.28	33.17	32.95	9.00	0	tlim	227.90	10.40	0.03	tlim	469.05
21	201	2781.75	149.83	149.66	2.70	0	tlim	455.10	2.81	0.03	tlim	188.24
22	213	1720.52	15.66	15.65	5.30	0	tlim	210.20	5.92	0	tlim	108.51
23	356	4001.71	tlim	600.92	3.80	1.58	tlim	448.00	6.25	3.81	tlim	547.32

TABLE 3.2: Saving comparison between manual and optimization solutions.

		Operator	Gurobi		HiGHS		CBC	
#	$\|J\|$	Obj. Value (€)	Obj. Value (€)	Saving (%)	Obj. Value (€)	Saving (%)	Obj. Value (€)	Saving (%)
1	7	296.91	295.07	0.62	295.07	0.62	295.07	0.62
2	10	305.98	301.14	1.58	301.14	1.58	301.14	1.58
3	30	361.05	332.63	7.87	332.63	7.87	332.63	7.87
4	50	582.67	352.24	39.55	352.24	39.55	352.24	39.55
5	53	707.73	666.75	5.79	666.75	5.79	666.75	5.79
6	70	716.64	471.40	34.22	471.40	34.22	471.40	34.22
7	81	1023.82	763.16	25.46	763.16	25.46	763.16	25.46
8	95	1216.47	941.47	22.61	941.47	22.61	941.47	22.61
9	100	1130.22	770.34	31.84	770.34	31.84	770.34	31.84
10	103	1289.91	900.52	30.19	900.52	30.19	900.52	30.19
11	106	4289.64	3983.02	7.15	3983.02	7.15	3983.43	7.14
12	107	1720.06	1234.02	28.26	1234.02	28.26	1234.02	28.26
13	108	1606.58	1102.63	31.37	1102.63	31.37	1102.63	31.37
14	115	1584.86	1204.98	23.97	1204.98	23.97	1204.98	23.97
15	121	1516.47	905.31	40.30	905.31	40.30	905.31	40.30
16	164	2713.36	1896.26	30.11	1982.65	26.93	1982.80	26.92
17	167	2436.15	1557.90	36.05	1557.90	36.05	1557.90	36.05
18	179	2783.97	2001.93	28.10	2003.51	28.03	2001.93	28.10
19	190	2719.10	1985.83	26.97	1985.83	26.97	1986.23	26.95
20	201	2870.88	2083.28	27.43	2083.28	27.43	2083.69	27.42
21	201	3712.04	2781.75	25.06	2782.32	25.04	2782.72	25.03
22	213	2573.06	1720.52	33.13	1720.52	33.13	1720.52	33.13
23	356	5607.28	4001.71	28.63	4012.19	28.45	4100.33	26.87

than linearly, due to the additional human interactions involved. Additionally, digitalization also reduces the time needed for sales representatives for email management. The estimated savings, considering the average email response time, amount to 30 hours per month. Overall, the digitalization of the process allows for a minimum saving of 120 hours monthly, which can be redirected to higher-value activities.

3.7 Conclusions

DSS are gaining increasing popularity within companies. This chapter outlines the creation of a model-driven DSS designed to address the challenges posed by intralogistics. In particular, the proposed DSS addresses a context with peripheral storage and centralized distribution, optional feature selection, and different stock unit configurations.

The DSS has been implemented to optimize both information and material flows. Regarding information, the process has been digitalized, eliminating repetitive and non-value-added information streams. This was made possible through a custom software architecture based on containers. Decisions regarding material flow have been optimized through an ILP model that determines the optimal choices for transferring goods from each warehouse and composing loads on transportation vehicles.

The proposed approach has been tested on real-world instances with different numbers of orders. Three different solvers were employed to evaluate the trade-off between Gurobi's superior performance and HiGHS and CBC's open-source licenses. Computational results were compared in terms of solutions and required time. Gurobi successfully solves nearly all instances relatively fast, while CBC and HiGHS usually achieve optimal values for the objective function, although without demonstrating optimality within the specified time limit. Overall, the results show a significant reduction in total costs compared to the company's manually calculated solution by operators. Furthermore, the digitalization of the process minimizes non-value-added time for both logistics and sales operators. Therefore, the implementation of the DSS offers economic benefits to the company by lowering expenses associated with stock transfers and gaining valuable working hours.

Nevertheless, further enhancements are possible. Currently, optimization occurs daily. Exploring optimization frequency via sensitivity analysis could balance economic gain and service level trade-offs. Less frequent optimization accumulates more orders, potentially improving margins. Yet, order accumulation delays shipments, reducing service levels.

Moreover, running the model for large instances can conflict with the company's needs due to significant time requirements. Since material quantities are updated only upon order consolidation and solution validation, sales operators using the system in real-time may concurrently request the same material, leading to resource contention. To address this issue, heuristic algorithms could be implemented to obtain good solutions in a limited amount of time. Alternatively, stochastic approaches could be adopted to anticipate material requests and mitigate resource contention under real-time usage.

Finally, the adoption of a model-driven DSS also has implications with respect to the replacement of human decision-making. The proposed DSS addresses the same operational problem traditionally handled by logistics operators, but within a formalized and repeatable optimization framework. While the model captures the main decision drivers, the problem is treated as offline and deterministic, in line with the current operational setting of the company. The DSS is therefore intended to support human operators by automating complex and repetitive decision tasks, rather than fully replacing human expertise.

References

ACIMAC Research Department (Oct. 2023). *World Production and Consumption of Ceramic Tiles*. Tech. rep. 11. Manufacturing Economics Study.

Alemany, MME et al. (2013). "Available-To-Promise modeling for multi-plant manufacturing characterized by lack of homogeneity in the product: An illustration of a ceramic case". In: *Applied Mathematical Modelling* 37.5, pp. 3380–3398.

Chai, Junyi, James NK Liu, and Eric WT Ngai (2013). "Application of decision-making techniques in supplier selection: A systematic review of literature". In: *Expert systems with applications* 40.10, pp. 3872–3885.

Chiou, Chuang-Chun (2008). "Transshipment problems in supply chain systems: review and extensions". In: *Supply Chain*, pp. 427–448.

Erdem, Aslı Sencer and Emir Göçen (2012). "Development of a decision support system for supplier evaluation and order allocation". In: *Expert Systems with Applications* 39.5, pp. 4927–4937.

Glickman, Theodore S and Susan C White (2008). "Optimal vendor selection in a multiproduct supply chain with truckload discounts". In: *Transportation Research Part E: Logistics and Transportation Review* 44.5, pp. 684–695.

Li, Yantong et al. (2020). "The multi-plant perishable food production routing with packaging consideration". In: *International Journal of Production Economics* 221, p. 107472.

Mendoza, Abraham and Jose A Ventura (2008). "An effective method to supplier selection and order quantity allocation". In: *International journal of business and systems research* 2.1, pp. 1–15.

Mishra, Anurag, Pankaj Dutta, and Naveen Gottipalli (2023). "An optimization model for the downstream supply chain network, considering consolidated warehouses and the selection of transportation mode". In: *International Journal of Productivity and Performance Management*.

Patil, Amitkumar et al. (2021). "Modeling Multi-Plant Capacitated Lot Sizing Problem with Interplant Transfer". In: *International Journal of Mathematical, Engineering and Management Sciences* 6.3, p. 961.

Pazhani, Subramanian, José A Ventura, and Abraham Mendoza (2016). "A serial inventory system with supplier selection and order quantity allocation considering transportation costs". In: *Applied Mathematical Modelling* 40.1, pp. 612–634.

Scott, James et al. (2015). "A decision support system for supplier selection and order allocation in stochastic, multi-stakeholder and multi-criteria environments". In: *International Journal of Production Economics* 166, pp. 226–237.

Zhang, Yuankai et al. (2021). "Multi-warehouse package consolidation for split orders in online retailing". In: *European journal of operational research* 289.3, pp. 1040–1055.

Chapter 4

Combining Learning and Heuristics for Pallet Prediction in Ceramics Tiles Distribution [1]

Motivated by a real-world case study in ceramic tile production, this chapter addresses the problem of determining the number of pallets required to load a given set of boxes. The problem must be solved quickly to give customers an expectation of the transportation cost of their orders. In addition, not all constraints and instance data can be easily determined in advance, and the items are loaded onto the pallets by operators who mostly rely on their personal experience. Therefore, traditional model-based solution methods do not apply well, and datadriven approaches are preferable. To solve the problem, we propose a hybrid algorithm in which a ML technique is trained over a company dataset comprising two years of customer orders, with the aim of predicting the number of pallets required by an order. The accuracy of the machine learning technique is largely improved by including additional features, such as lower and upper bounds, in the dataset, obtained using quick optimization algorithms. The resulting hybrid algorithm has been compared with the model-based software currently used at the company, consistently providing better-quality results in shorter computing times.

4.1 Introduction

The global ceramic tile market has grown significantly over the past decade, with worldwide production reaching 16.8 billion square meters (ACIMAC Research Department, 2023). Italy is a leading exporter, generating €7.2 billion in revenue, highlighting the sector's importance to the national economy. Efficient logistics optimization in the ceramic tile industry, including loading of tiles on pallets and transportation of pallets, is crucial for cost reduction and company competitiveness. Loading operations involve transferring specified quantities of ceramic tile boxes, either of the same size or different sizes, from production pallets to shipping pallets to fulfill customers' orders. When loading pallets, the company needs to carefully consider weight and volume constraints and keep pallet stability and balance to ensure the safe transport of tiles while using the minimum number of shipping pallets.

In this chapter, we handle a variant of the distributor's pallet loading problem (DPLP) (Mungwattana et al., 2023) that emerges from an international ceramic tile company headquartered in Italy. The company has an e-commerce platform to receive customers' orders. As the company does not provide the transportation of the shipping pallets, customers need to collect their pallets by using their vehicles or third-party carriers. Consequently, the company

[1] The results of this chapter appears in : Taccini, M., de Oliveira, M. A., dos Santos, A.G., de Queiroz, T. A. and Iori, M. "A Hybrid Approach for Pallet Loading in Ceramic Tile Industry". In *Proceeding of the International Conference on Optimization and Learning, 2025, pp. 329-340.*

should provide the number of shipping pallets associated with each order or at least predict this number accurately, especially for customers who need to contract a third-party carrier. This prediction is complex, even if using specialized software to address pallet loading problems (Silva et al., 2016), because the company faces specific constraints in its operational daily routine, necessitating extensive customization of such software.

In this chapter, we propose a hybrid approach to predict the number of shipping pallets required for a given customer order. The approach combines ML techniques to gain information from a large dataset of previously shipped pallets with techniques from the optimization field to enrich the dataset by quickly providing additional features such as lower and upper bounds on the number of pallets. The hybrid approach is then compared with solutions currently in use at the company, including manual operators' decisions and outputs from a given specialized software (PackVol, 2024). The computational experiments show that the proposed approach is a valuable tool for the company.

The remainder of the chapter is organized as follows. Section 4.2 contains a brief review of the literature related to the DPLP. Section 4.3 describes the company problem with its constraints and requirements. Section 4.4 presents our proposed hybrid approach. Section 4.5 contains the numerical experiments on real-world instances. Finally, Section 4.6 concludes the chapter and provides several directions for future research.

4.2 Literature Review

The literature on cutting and packing problems is extensive. According to Sweeney and Paternoster (1992), the first scientific papers on these problems date back to the 1940s. The most recent surveys concern problems in one-, two-, and three-dimensions and contain detailed discussions on solution methods based on exact and heuristic approaches (see, e.g., Iori et al. (2021a) and Yagiura et al. (2025)). An essential characteristic of these problems is given by the number of practical constraints they may include, such as complete shipment, conflicting items, cargo stability, load-bearing, multi-drop, and load-balancing (Nascimento et al., 2021). Concerning pallet loading problems, Silva et al. (2016) surveyed the manufacturer's variant, in which all items are identical, and the objective is to pack the maximum number of items onto a single pallet. Besides commenting on the different data sets and solution methods, the authors emphasized that the computational complexity of this problem is still an open question. The DPLP, on the other hand, assumes that items are not necessarily identical and is an NP-hard problem. Its objective is to find the minimum number of pallets necessary to load all items. The DPLP is also referenced as the multi-pallet loading problem (Terno et al., 2000) and the pallet building problem (Calzavara et al., 2021).

One of the first authors to handle the DPLP was Hodgson (1982), considering the transport of palletized cargo by the U.S. Air Force. The author proposed a dynamic programming-based algorithm and tested it on instances having 30 boxes. In Terno et al. (2000), the DPLP was solved using an algorithm based on branch-and-bound. The authors considered practical constraints such as weight limits, load-bearing, stability, and grouping of items. In Birgin et al. (2012), the problem had no limit on the number of boxes to load, and boxes could have multiple orientations. The authors developed a recursive partitioning algorithm, which integrated a recursive five-block heuristic and an L-approach. The authors could not find an instance for which this algorithm failed in finding an optimal solution (such an instance was later found in Queiroz et al. (2015)). In Calzavara et al. (2021), the DPLP emerged from a robotized task of loading boxes into layers, which were then stacked while meeting load-bearing requirements. Constraints like contiguity, visibility, and multiple orientations were considered. The authors proposed a mathematical model and a reactive greedy randomized adaptive search procedure for solving realistic, large-sized instances. A new DPLP variant in which pallets may have different sizes was handled in Mungwattana et al. (2023). The problem emerged from a lamp and lighting manufacturing company that needed to load many carton boxes onto pallets of multiple sizes. Constraints such as load-bearing and multiple orientations were taken into consideration.

Using artificial intelligence techniques to handle combinatorial optimization problems is a promising area of research. The survey in Bengio et al. (2021) discussed recent advances in combining ML techniques with optimization methods to efficiently handle hard combinatorial optimization problems. The authors also commented on the benefits (e.g., speed and generalization) and challenges (e.g., training and accuracy) of using ML purely. Concerning cutting and packing problems, an increasing number of publications combine ML or artificial intelligence techniques with exact or heuristic algorithms to handle, e.g., bin packing (Fang et al., 2023) and container loading problems (Que et al., 2023).

4.3 Problem Description

The problem under investigation is a variant of the DPLP with specific operational and product-related constraints. After being produced, the ceramic tiles are first loaded on production pallets, which are only used for inbound logistics. An automated system then delivers the production pallets to the loading area. The company operators load the products onto shipping pallets using hydraulic machines. The automated system is also responsible for other logistics activities inside the warehouse. As a result, products are not available simultaneously, as their arrival sequence depends on the system's decisions and the warehouse's current state. For instance, machinery downtime may alter the order in which production pallets are sent to the loading areas. Moreover, the loading area cannot have more than four production pallets at a time. Due to this space limitation, operators need to load the products onto the pallets as soon as possible. As other products arrive, operators continue loading until the shipping pallets of that order are completed before moving to the next order. When loading a shipping pallet, operators must also satisfy load requirements such as vertical stability, load-bearing, and restricted box orientations.

Our DPLP can be formally described as follows. A customer order consists of a set B of three-dimensional boxes, each containing a given number of tiles. Each box belongs to a type $t \in T = \{1, 2, ..., m\}$, corresponding to a specific tile size. A box of type t has width $w_t \in \mathbb{Z}_+^*$, depth $l_t \in \mathbb{Z}_+^*$, height $h_t \in \mathbb{Z}_+^*$, and weight $p_t \in \mathbb{Z}_+^*$. Let $d_t \in \mathbb{N}$ denote the number of boxes of type t required by the order. Let P represent the set of identical shipping pallets, each with a two-dimensional loading surface of width $W \in \mathbb{Z}_+^*$ and depth $L \in \mathbb{Z}_+^*$. The DPLP's objective is to load all boxes requested in the order by using the minimum number of pallets. Constraints on the maximum pallet height $H \in \mathbb{Z}_+^*$ and weight $W_{\max} \in \mathbb{Z}_+^*$ need to be satisfied.

Due to these complex company constraints, obtaining a solution for this DPLP variant is not easy, especially for large-sized and heterogeneous orders (i.e., composed of boxes of different sizes). Even if a DPLP solution satisfying all constraints is available, it could suffer from warehouse uncertainties, possibly making it impractical for the company operators. Besides that, the company should give the number of pallets associated with each customer's order at the moment of purchase on its e-commerce platform. Therefore, the company is currently interested in a fast and accurate tool capable of predicting the number of shipping pallets, rather than solving a dedicated but time-consuming optimization problem.

4.4 Hybrid Model

We propose a hybrid approach to the DPLP under investigation that considers optimization algorithms to enhance the predictive performance of an ML model. The company provided data on two years of sales and the corresponding number of pallets generated for each order. This data includes the maximum number of boxes per shipping pallet for each box size t, denoted by $F_t^{\max}$, and the maximum weight ($W_{\max}$) and volume (WLH) a shipping pallet can support.

An initial training dataset is created from the company-provided data. We start calculating the frequency $F_t = d_t$ (i.e., the number of boxes of type t), the weight $W_t = d_t p_t$, and the volume $V_t = w_t l_t h_t d_t$ for each type $t \in T$. The dataset is labeled with the number of shipping pallets Z_j generated by each order $j \in \{1, ..., n\}$. Since training on the initial dataset can take a long time, all intermediate training steps (Sections 4.4.1 - 4.4.3) are conducted using a reduced dataset. The complete dataset is only used on the final computational experiments (Section 4.5). The reduced dataset is created by randomly selecting a subset of orders from the initial dataset, resulting in a training set with about $\frac{1}{8}$ of the total number of orders.

4.4.1 Machine Learning Model Selection

We developed a pipeline to train and test four different machine learning models. These models, which were chosen based on the literature review and our experience, consist of: XGBoost (Chen and Guestrin, 2016a), LightGBM (LGBM) (LightGBM, 2024), Random Forest, and Support Vector Machine (SVM) from scikit-learn (Pedregosa et al., 2011). Each model passes through a hyperparameter tuning phase on the Mean Squared Error (MSE) and Mean Absolute Error (MAE) metrics, obtained by performing a standard grid search. Table 4.1 reports the hyperparameters and their possible values, besides the best values found along with MSE and MAE metrics. The possible values were defined using a trial-and-error approach and following the indications in each model documentation and the literature (see, e.g., Banerjee (2020)).

TABLE 4.1: Grid search results with the hyperparameter values and errors.

Model	Hyperparameter	Values	Best Value	MSE	MAE
XGBoost	learning_rate max_depth gamma	0.1, 0.3 4, 5, 6 0, 0.25, 1	0.3 4 0.25	4.78	1.17
LGBM	learning_rate num_leaves	0.005, 0.01 6, 8, 12, 16	0.01 16	16.49	2.58
Random Forest	max_features max_depth max_leaf_nodes	sqrt, log2, None 3, 6, 9 3, 6, 9	None 9 9	13.61	2.26
SVM	C gamma	0.1, 1, 10, 100, 1000 1, 0.1, 0.01, 0.001, 0.0001	100 0.0001	35.89	3.45

The results in Table 4.1 show that XGBoost outperforms the other models regarding the MSE and MAE values. Consequently, we decided to use only XGBoost in the following steps. We also decided to perform a deeper tuning of this model's hyperparameters. Table 4.2 contains the outcome of this additional tuning test, which leads to better MSE and MAE values for the XGBoost model.

TABLE 4.2: XGBoost tuning with the best hyperparameter values and errors.

Hyperparameter	Values	Best Value	MSE	MAE
learning_rate	0.01, 0.1, 0.2, 0.3	0.3	1.11	0.43
gamma	0, 0.1, 0.5	0.1		
max_depth	3, 6, 10	6		
min_child_weight	0, 1, 3, 5	0		
subsample	0.5, 1	0.5		
colsample_bytree	0.5, 1	1		

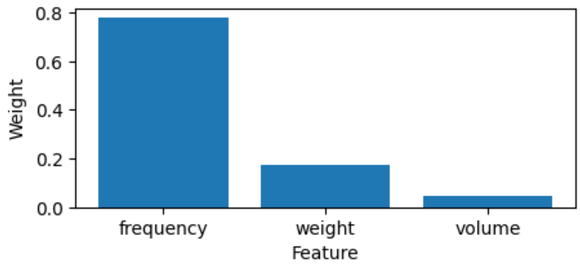


FIGURE 4.1: Importance of the feature weights in the XGBoost model.

4.4.2 Features Optimization

After successfully fine-tuning the XGBoost model, we investigated the importance of each feature in the reduced dataset. This analysis reveals that columns W_t and V_t, for $t \in T$, are marginally contributing to the model's predictions, as shown in Figure 4.1. Furthermore, tile sizes that appear less frequently in the dataset exhibit higher error rates, as shown in Figure 4.2.

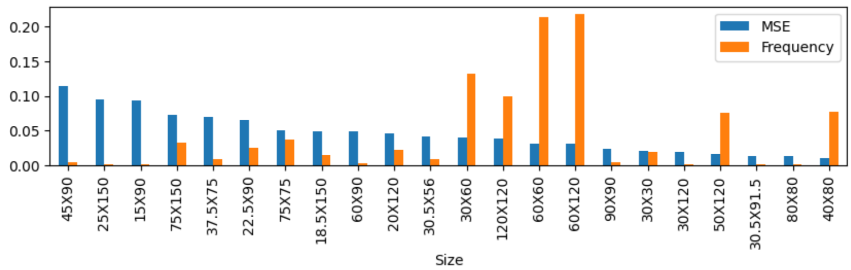


FIGURE 4.2: MSE and frequency values (F_t), normalized and sorted by decreasing MSE.

These results allow the removal of less informative features, such as the size-grouped weight W_t and volume V_t. At the same time, we introduce more general order-aggregated features, as follows. Let O be the set of orders we need to process, and let $o \in O$ be a given order. We determine the total number of boxes $F_o = \sum_{t=1}^{m} F_t$, the total weight $W_o = \sum_{t=1}^{m} W_t$, the total volume $V_o = \sum_{t=1}^{m} V_t$, the average box weight $W_{avg} = \frac{W_o}{F_o}$, and the average box volume $V_{avg} = \frac{V_o}{F_o}$. Figure 4.3 shows how these new features impacted the predictive capability of the XGBoost model.

In the set O of orders, we observe that predicting the number of shipping pallets for homogeneous orders, defined O_h, which consist of only one type of box, and O_{full} ones, which require only production pallets regardless of the types, is straightforward for the model. In this way, we divided results into two groups: one containing all available orders O and another excluding these simpler orders $O_{mix} = O - (O_h \cup O_{full})$. With this strategy, we evaluated

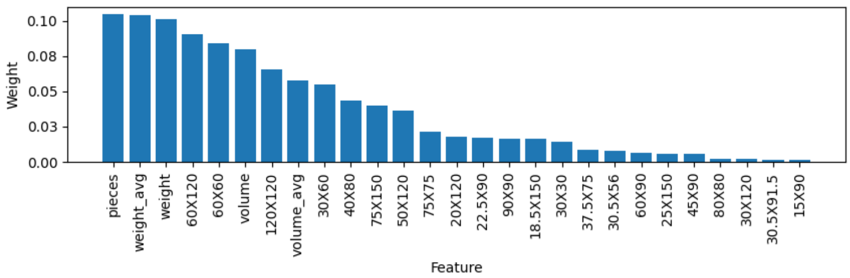


FIGURE 4.3: Importance of the new order-aggregated features in the XG-Boost model.

the model in both scenarios, preventing simple orders to improve the metrics and resulting in premature conclusions. Table 4.3 summarizes the model's performance before and after including the new features.

TABLE 4.3: MSE and MAE values for each group of orders and the dataset structures S_1 (original) and S_2 (enriched with additional features).

	MSE		MAE	
Dataset structure	O	O_{mix}	O	O_{mix}
S_1	1.01	1.38	0.42	0.53
S_2	0.36	0.47	0.24	0.31

4.4.3 Incorporation of Optimization-based Features

Aiming to reduce the errors further, three additional features (i.e., HS_1, HS_2, and HS_3) computed with quick algorithms are introduced into the training dataset. To ensure computational efficiency, the proposed algorithms have the worst-case time complexity of at most $O(F_o^2)$, where F_o is the total number of boxes in the customer's order. The proposed algorithms are as follows:

- HS_1: calculated as $HS_1 = \max\left(\lceil\frac{\sum_{t=1}^{m} W_t}{W_{\max}}\rceil, \lceil\frac{\sum_{t=1}^{m} V_t}{WLH}\rceil\right)$. This can be interpreted as the value of the linear relaxation lower bound (LB);
- HS_2: calculated as $HS_2 = \sum_{t=1}^{m}\lceil\frac{F_t}{F_t^{\max}}\rceil$, where $F_t^{\max}$ is the maximum number of boxes of type t that can be loaded onto a pallet. This is, instead, an upper bound (UB) for the DPLP instance, computed by considering that all pallets are built with only homogeneous tiles (and this can easily be done by satisfying vertical stability and load bearing constraints);
- HS_3: described in Algorithms 1 and 2. This algorithm loads pallets by stacking layers of same-sized boxes, starting with those having the largest area until the pallet limits are reached. Once full, it begins a new pallet and continues until all boxes are packed. This may be seen as a quick heuristic solution approach, but, in fact, it does not consider vertical stability and load bearing constraints, and hence it only provides an approximation function of the minimum number of pallets.

Algorithm 1 HS3 - constructive heuristic to give the number of shipping pallets

$pallets \leftarrow 1$ ▷ Pallet estimation
for $t \in \text{SORTED}(T)$ **do** ▷ Iterate over box types sorted by decreasing base area
  while $d_t > 0$ **do** ▷ Boxes of type t left to pack
    if $w - p_t < 0$ **OR** $h - h_t < 0$ **OR** no space to pack one box **then**
      $x, y, w, h \leftarrow W, L, W_{\max}, H$ ▷ Restart using an empty pallet
      $pallets \leftarrow pallets + 1$
    end if
    $b \leftarrow \lfloor \frac{w}{p_t} \rfloor$ ▷ How many boxes can be packed based on weight
    $q, d_x, d_y \leftarrow \text{TRYPACK}(b, w_t, l_t, x, y)$ ▷ Try to pack directly
    $q_2, d_{x2}, d_{y2} \leftarrow \text{TRYPACK}(b, l_t, w_t, x, y)$ ▷ Try rotating to pack
    if $q < q_2$ **OR** $(q = q_2$ **AND** $p_x p_y < p_{x2} p_{y2})$ **then**
      $q, d_x, d_y \leftarrow q_2, d_{x2}, d_{y2}$ ▷ Choose to pack more boxes, then more area
    end if
    $d_t, w, h \leftarrow d_t - q, w - p_t q, h - h_t$ ▷ Pack boxes, update weight and height
    $x, y \leftarrow d_x, d_y$ ▷ Update dimensions for next layer
  end while
end for
return $pallets$

Algorithm 2 TRYPACK

Input: b, b_x, b_y, l_x, l_y ▷ Boxes to pack, box and layer dimensions
$X \leftarrow \min(b, \lfloor \frac{l_x}{b_x} \rfloor)$ ▷ Maximum number of boxes in x-dimension
$Y \leftarrow \min(b, \lfloor \frac{l_y}{b_y} \rfloor)$ ▷ Maximum number of boxes in y-dimension
$q, d_x, d_y \leftarrow 0, 0, 0$ ▷ Best quantity and new layer dimensions
for $x \in \{1 \ldots X\}$ **do** ▷ Test all number of boxes in x-dimension
  $y \leftarrow \min(Y, \lceil \frac{b}{x} \rceil)$ ▷ How many rows in y-dimension will be needed
  if $q < \min(b, xy)$ **OR** $(q = \min(b, xy)$ **AND** $d_x d_y < b_x b_y xy)$ **then**
    $q, d_x, d_y \leftarrow \min(b, xy), b_x x, b_y y$
  end if
end for
return q, d_x, d_y

Table 4.4 describes the final structure S_3 of the training dataset, which now includes the three additional optimization-based features. As shown in Table 4.5, the MSE and MAE values for the dataset structure S_3 are 0.24 and 0.19 for O, and 0.32 and 0.26 for O_{mix}, respectively, representing a considerable improvement in the accuracy of the model compared to previous S_2 structure.

TABLE 4.4: Final training dataset structure S_3.

j	F_1	...	F_m	F_o	W_o	V_o	W_{avg}	V_{avg}	HS_1	HS_2	HS_3	Z_j
1	10	...	3	25	250	500	10	20	5	9	8	7
...	...	...	...	...	...	...	...	...	...	...	...	...
n	7	...	0	13	195	65	15	5	2	4	3	3

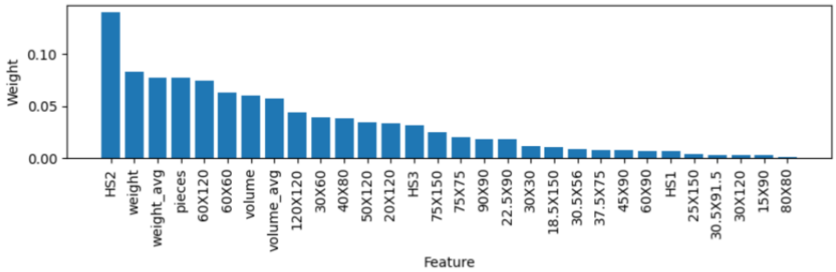


FIGURE 4.4: Importance of HS_1, HS_2, HS_3 features in the XGBoost model.

TABLE 4.5: MSE and MAE values for each group of orders and the dataset structures S_1 (original), S_2 (enriched with additional features) and S_3 (added optimization-based features).

	MSE		MAE	
Dataset structure	O	O_{mix}	O	O_{mix}
S_1	1.01	1.38	0.42	0.53
S_2	0.36	0.47	0.24	0.31
S_3	0.24	0.32	0.19	0.26

Moreover, as show in Figure 4.4, the newly introduced optimization-based features exhibit high weight values, especially the HS_2 feature, which attains the highest values among all features demonstrating that optimization-based features are relevant for the model prediction and provide a major contribution to error reduction.

4.5 Computational Experiments

All the codes were implemented in Python 3.10 and executed in a computer with an Intel Core i5-1135G7 2.40 GHz processor, 8 GB of RAM, and Linux Mint 21 Cinnamon as the operating system. We evaluate the XGBoost model considering the structure S_3 (i.e., the one enriched with the additional optimization-based features) and the complete two-year dataset with all orders. This corresponds to our hybrid approach. In Table 4.6, we observe the impact of the two groups O_h and O_{full} on the training phase. Consequently, we train the model using the structure S_3, first including all orders O and then keeping only mixed orders O_{mix} (which we recall is equal to $O - (O_h \cup O_{full})$).

TABLE 4.6: MSE and MAE values for each group of orders and the dataset structure S_3.

			MSE		MAE	
Groups of orders	Orders	Training Time (min)	O	O_{mix}	O	O_{mix}
O	166391	54.79	0.22	0.73	0.11	0.33
O_{mix}	40533	26.90	0.25	0.76	0.12	0.33

As reported in Table 4.6, applying the model to all orders O yields slightly better results compared to using O_{mix}. However, using only O_{mix} can reduce the training time by around 51% while keeping satisfactory predictive results.

We next compare in Table 4.7 the proposed hybrid approach with PackVol (PackVol, 2024), a specialized software the company uses to solve the DPLP heuristically. PackVol was configured by the company employees to satisfy the specific operational constraints of the DPLP as closely as possible. This means that PackVol is not able to capture some constraints. For example, as mentioned in Section 4.3, ceramic tiles of a given order are not all available simultaneously due to the warehouse's current state. As a result, PackVol could underestimate the number of pallets. We also present the number of pallets obtained from the company operators for comparison purposes. This experiment considers 30 real instances, consisting of 30 orders randomly selected among the heterogeneous orders made in 2024 (which are not included in the training set). The real instances I are all heterogeneous, and are sorted by increasing number of box types m. Its maximum value is 6 since 99.32% of the orders in the training dataset contain six or fewer types. Table 4.7 contains the instance number and the solution value (i.e., number of shipping pallets) obtained with PackVol, the proposed hybrid approach, and the company operators. We assume the operators' solution is the basis for comparison, so the columns "Diff." contain the difference with such a solution for each instance. Positive values indicate the usage of more pallets, which could be acceptable. Negative values indicate fewer pallets in the solution and could represent additional logistic issues for customers (e.g., the need for contracting "on the flight" an extra vehicle).

TABLE 4.7: Results for the 30 real instances and comparison among the different methods.

Instance	# types	Company	PackVol	Diff. (PackVol)	Hybrid	Diff. (Hybrid)
1	2	2	1	-1	2	0
2	2	6	6	0	7	1
3	2	4	4	0	4	0
4	2	1	1	0	1	0
5	2	6	6	0	6	0
6	2	2	2	0	2	0
7	3	22	22	0	22	0
8	3	5	5	0	5	0
9	3	5	3	-2	5	0
10	3	8	8	0	9	1
11	3	10	10	0	11	1
12	3	7	6	-1	7	0
13	4	5	4	-1	6	1
14	4	8	6	-2	7	-1
15	4	9	8	-1	9	0
16	4	7	7	0	7	0
17	4	15	15	0	16	1
18	4	6	6	0	7	1
19	5	26	25	-1	26	0
20	5	9	8	-1	9	0
21	5	28	27	-1	27	-1
22	5	2	4	2	3	1
23	5	13	13	0	13	0
24	5	10	6	-4	10	0
25	6	22	18	-4	21	-1
26	6	10	8	-2	10	0
27	6	22	21	-1	22	0
28	6	21	19	-2	20	-1
29	6	10	10	0	12	2
30	6	16	14	-2	16	0

Observing the results in Table 4.7, PackVol achieves an MSE of 2.13 and an MAE of 0.93, whereas our hybrid approach achieves significantly better results, with 0.50 and 0.43 values of MSE and MAE, respectively. These outcomes highlight that our approach outperforms PackVol in both metrics, demonstrating its capability to predict the number of pallets following the real context of the company with its operational requirements and workflows. Another important aspect is the computation time required by the proposed approach. Although the training phase may be time-consuming, after training, it solved all instances in just 0.016 seconds, while PackVol required around one second per instance. From Table 4.7, we can also observe that PackVol returns solutions different from those by the company in 16 cases, returning fewer pallets for 15 cases and achieving a largest deviation of -4 pallets. This may bring issues for customers. On the other hand, the hybrid approach returns solutions that differ from the company ones for just 12 instances, returning fewer pallets for only 4 of them and having the largest deviation of just 2 pallets. These values confirm that our approach, trained on real-world data, effectively models the problem constraints and results in accurate predictions.

4.6 Concluding Remarks

We proposed a hybrid approach composed of machine learning and optimization components to predict the number of pallets required to ship an order of ceramic tiles. We evaluated four machine learning models, and after fine-tuning, we selected XGBoost. Its performance was improved by adding new features that capture general information about the company's operational constraints. We also incorporated three measures obtained by quick optimization algorithms, leading to significant reductions in both MSE and MAE values. The resulting hybrid approach obtained better results than PackVol, the specialized model-based software currently used at the company.

The computational experiments highlight that our hybrid approach, once it is trained on real-world data, is fast and well-suited for the company-specific operational constraints. It allows more accurate and practical predictions than PackVol, resulting in more solutions equal to the ones obtained by the company operators and a reduced number of solutions requiring fewer pallets. This is remarkably achieved without feeding the approach with any direct information about the pallet loading constraints (i.e., stability, load-bearing requirements, and orientation), making the approach a very flexible tool.

Future research could propose new features to the hybrid approach (e.g., solutions obtained with heuristics based on local search). Another interesting research avenue is to work on a more general problem, considering pallets of different sizes, which will lead to a more challenging objective function. We finally plan to test the approach to other related pallet loading, cutting, and packing problems, as well as scheduling problems (Borges et al., 2024; Queiroz and Mundim, 2020).

References

ACIMAC Research Department (Oct. 2023). *World Production and Consumption of Ceramic Tiles*. Tech. rep. 11. Manufacturing Economics Study.

Banerjee, Prashant (2020). *A Guide on XGBoost hyperparameters tuning*. https://www.kaggle.com/code/prashant111/a-guide-on-xgboost-hyperparameters-tuning. Accessed: 2024-09-26.

Bengio, Yoshua, Andrea Lodi, and Antoine Prouvost (2021). "Machine learning for combinatorial optimization: a methodological tour d'horizon". In: *European Journal of Operational Research* 290.2, pp. 405–421.

Birgin, E. G., R. D. Lobato, and R. Morabito (2012). "Generating unconstrained two-dimensional non-guillotine cutting patterns by a Recursive Partitioning Algorithm". In: *Journal of the Operational Research Society* 63.2, pp. 183–200.

Borges, Yulle G. F. et al. (Oct. 2024). "Algorithms for the bin packing problem with scenarios". In: *Journal of Combinatorial Optimization* 48.4, p. 34.

Calzavara, Gabriele et al. (2021). "Mathematical models and heuristic algorithms for pallet building problems with practical constraints". In: *Annals of Operations Research*. ISSN: 1572-9338.

Chen, Tianqi and Carlos Guestrin (2016a). "XGBoost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.

Fang, Jie et al. (2023). "A Hybrid Reinforcement Learning Algorithm for 2D Irregular Packing Problems". In: *Mathematics* 11.2. ISSN: 2227-7390.

Hodgson, Thom J. (1982). "A Combined Approach to the Pallet Loading Problem". In: *A I I E Transactions* 14.3, pp. 175–182.

Iori, Manuel et al. (2021a). "Exact solution techniques for two-dimensional cutting and packing". In: *European Journal of Operational Research* 289.2, pp. 399–415. ISSN: 0377-2217.

LightGBM (2024). *Light Gradient Boosting Machine*. https://github.com/microsoft/LightGBM. Accessed: 2024-12-10.

Mungwattana, Anan, Thanakrit Piyachayawat, and Gerrit K. Janssens (2023). "A two-step evolutionary algorithm for the distributor's pallet loading problem with multi-size pallets". In: *Flexible Services and Manufacturing Journal* 35.4, pp. 1256–1275.

Nascimento, Oliviana Xavier, Thiago Alves Queiroz, and Leonardo Junqueira (2021). "Practical constraints in the container loading problem: Comprehensive formulations and exact algorithm". In: *Computers & Operations Research* 128, p. 105186. ISSN: 0305-0548.

PackVol (2024). *Optimization Software for Load Planning*. https://www.packvol.com/index.html. Accessed: 2024-12-10.

Pedregosa, Fabian et al. (2011). "Scikit-learn: Machine learning in Python". In: *the Journal of machine Learning research* 12, pp. 2825–2830.

Que, Quanqing, Fang Yang, and Defu Zhang (2023). "Solving 3D packing problem using Transformer network and reinforcement learning". In: *Expert Systems with Applications* 214, p. 119153. ISSN: 0957-4174.

Queiroz, T. A., F. K. Miyazawa, and Y. Wakabayashi (June 2015). "On the L-approach for generating unconstrained two-dimensional non-guillotine cutting patterns". In: *4OR* 13.2, pp. 199–219.

Queiroz, Thiago Alves and Leandro Resende Mundim (2020). "Multiobjective pseudo-variable neighborhood descent for a bicriteria parallel machine scheduling problem with setup time". In: *International Transactions in Operational Research* 27.3, pp. 1478–1500.

Silva, Elsa, José F Oliveira, and Gerhard Wäscher (2016). "The pallet loading problem: a review of solution methods and computational experiments". In: *International Transactions in Operational Research* 23.1-2, pp. 147–172.

Sweeney, Paul E. and Elizabeth Ridenour Paternoster (1992). "Cutting and Packing Problems: A Categorized, Application-Orientated Research Bibliography". In: *Journal of the Operational Research Society* 43.7, pp. 691–706.

Terno, Johannes et al. (2000). "An efficient approach for the multi-pallet loading problem". In: *European Journal of Operational Research* 123.2, pp. 372–381. ISSN: 0377-2217.

Yagiura, Mutsunori, Shinji Imahori, and Yannan Hu (2025). *Cutting and Packing Problems. From the Perspective of Combinatorial Optimization*. Springer Tokyo.

Chapter 5

Multi-Output Pallet Prediction in Ceramics Tiles Distribution [1]

Based on the study developed in Taccini et al. (2025), this chapter presents a ML approach that integrates optimization-derived features and multi-output regression strategies. Three strategies are developed: one based on independent regressors and two based on regressor chains. Computational experiments on real instances demonstrate that all proposed strategies outperform the company's solution. Among them, one of the regressor chain-based strategies achieves the best overall performance, offering the lowest error rates. The results highlight that combining ML with optimization provides accurate, scalable, and practical solutions for dynamic industrial environments. This study contributes to bridging the gap between ML and operations research for solving complex, highly constrained logistics problems.

5.1 Introduction

The ceramic tile industry has experienced steady global growth in recent years, with worldwide production reaching 16.8 billion square meters (ACIMAC Research Department, 2023). Italy ranks among the top exporters, generating €7.2 billion in revenue, which highlights the strategic role of the sector in the national economy. The management of pallet loading operations and transportation is an important component of competitiveness in this industry. These operations involve transferring boxes of ceramic tiles, which often vary in size and weight, from production pallets (used in internal logistics operations) to shipping pallets (used for external logistics) to fulfill customer orders. To ensure safe transportation and minimize logistics costs, each pallet must comply with specific weight and volume limits and maintain adequate structural integrity.

This study addresses a real-world pallet loading problem faced by a major international ceramic tile company headquartered in Italy. Customers' orders are submitted through the company's e-commerce platform or via standard commercial channels. Each customer is responsible for collecting the ceramics, either using their vehicle or relying on a third-party carrier. Consequently, the company must inform the customer in advance of both the number of pallets required for each order and the types of pallets that will be used. Since pallet types vary in size and capacity, they impact transportation planning and cost estimation for the customer.

To provide customers with this information, the company must solve a variant of the distributor's pallet loading problem (DPLP) (Silva et al., 2016), aiming to minimize the number of pallets required for loading boxes. Providing an optimal solution for such a problem is

[1] The results of this chapter appears in : Taccini, M., de Oliveira, M. A., dos Santos, A.G., de Queiroz, T. A. and Iori, M. "Combining Learning and Heuristics for Pallet Prediction in Ceramics Distribution". In *Proceeding of the International Conference on Optimization and Decision Science, 2025, forthcoming.*

not a straightforward task, as it involves not only its computational complexity (i.e., an NP-hard problem), but also numerous constraints and uncertainties arising from the company's warehouse and daily practices (e.g., not all production pallets are available at the same time to build the shipping pallets). Such limitations underscore the need for a tailored solution method that can learn from historical data and provide accurate, real-time predictions for the customers' orders.

In this work, we propose an approach that predicts both the quantity and types of shipping pallets required to fulfill a customer's order. It integrates machine learning with optimization heuristics to obtain accurate predictions, extending the preliminary study in Taccini et al. (2025). Its results are compared with the current solutions proposed by the company. These include decisions made by the warehouse's expert operators as well as the output produced by a specialized pallet loading software tool (PackVol, 2024). The results of the computational experiments demonstrate that the proposed approach provides practical benefits within the company's logistics processes. Additionally, this study contributes to the literature by bridging the gap between machine learning and operations research to solve a complex, highly constrained logistics problem in a dynamic industrial environment.

The structure of the chapter is as follows. In Section 5.2, we briefly describe the literature about the DPLP and related problems. In Section 5.3, we formally present the problem under investigation. In Section 5.4, we describe the proposed approach and detail each of its parts. In Section 5.5, we present and discuss the computational experiments on the company's instances. In Section 5.6, we conclude our research by listing the main findings and providing directions for future studies.

5.2 Literature Review

There are different variants of pallet loading problems. These are usually classified as cutting and packing problems, for which we refer to the typology proposed by Wäscher et al. (2007). General surveys on cutting and packing problems were provided by Hadj Salem et al. (2023) and Iori et al. (2021b), while real-world constraints have been discussed in works such as Bortfeldt and Wäscher (2013) and Nascimento et al. (2021). Pallet loading problems can be categorized into two main variants: those involving manufacturers and those involving distributors. The survey by Silva et al. (2016) addressed the manufacturer's pallet loading problem. This variant assumes that all items are identical and seeks to maximize the number of items packed onto a single pallet.

The problem we address here is a generalization of the distributor's pallet loading problem (i.e., the DPLP), also known in the literature as the pallet building problem and the multi-pallet loading problem. It assumes that one or more pallets are used to pack a set of items of various types, thereby minimizing the number of pallets required.

The number of contributions on the DPLP in the literature is significant. The earliest works date back to the 1980s, such as Hodgson (1982), which proposed an algorithm based on dynamic programming. In Bischoff et al. (1995), a heuristic that packs items in layers was introduced, allowing the inclusion of practical constraints such as vertical stability. Computational experiments yielded promising results; however, performance declined in instances with greater variability in box dimensions. In Queiroz et al. (2015), the authors reviewed previous algorithms, including that of Birgin et al. (2012), and presented instances where such methods fail to find optimal solutions. In Gzara et al. (2020), the authors addressed a DPLP for a logistics company requiring the packing of items onto pallets every two minutes in an automated warehouse processing thousands of items and hundreds of pallets daily. They proposed a layer-based column generation algorithm that incorporates constraints such as vertical stability, load-bearing capacity, pallet weight limits, and planogram sequencing.

However, the load-bearing constraint was modeled using a simplified load transfer based on surface area percentages.

Another practical DPLP was addressed in Calzavara et al. (2021), involving a robotized application with constraints including load-bearing, contiguity, visibility, and multiple orientations. Most recently, Yao et al. (2025) introduced a DPLP variant that incorporates a buffer area, which is useful for dealing with uncertainty related to partial information about items that becomes available only before packing. They proposed a greedy heuristic combining block generation, open space representation, and a selection strategy to improve pallet utilization. The resulting effectively balances efficiency and real-time decision-making, accounting for robot interference and practical constraints such as vertical stability.

Regarding the use of artificial intelligence techniques for pallet loading problems, Layeb and Omri (2024) solved instances of the manufacturer's variant using deep reinforcement learning, achieving an average volume utilization of approximately 99%. In contrast, Magnani et al. (2025) addressed the DPLP with practical constraints such as vertical stability and load-bearing. They proposed a hybrid approach that combines a heuristic, which constructs layers with items, with machine learning techniques (e.g., random forest and support vector regression) to identify promising layers for pallet construction.

5.3 Problem Description

The problem addressed in this study is a DPLP variant obtained from a large-scale ceramic tile manufacturer in Italy. We are given a set of customer orders. Each order requires a set of boxes. The boxes have three dimensions, and a demand is associated with each box. To supply these orders, the company has been using two types of shipping pallets: the first with a larger base area and higher load capacity, while the second is smaller and used for lighter or more compact products. The boxes are loaded in levels, respecting the pallet's rectangular base, height limit, and weight capacity, as well as its vertical stability, load-bearing requirements, and the allowed orientations of the boxes. There are also compatibility constraints, meaning that not both pallets can be used to transport all types of boxes. The problem's objective is to load all boxes of the given order using the minimum quantity of pallets, while satisfying all the previously mentioned constraints.

Additional details about the company's warehouse and loading operations are provided below to better situate the problem. Ceramic tiles are first loaded onto standardized production pallets, which are not used for shipping but act as intermediate supports during internal logistics transport. These pallets are transported to the loading area via an automated material handling system, which has a limited capacity. Consequently, production pallets cannot be simultaneously available for the company's operators during the loading of shipping pallets. Their arrival is influenced by the scheduling choices of the warehouse's automated system and its current operational status. For instance, machinery failures can delay pallet delivery, disrupting the sequence in which production pallets reach the loading area, which itself faces strict spatial limitations. As a result, operators must begin transferring boxes from each production pallet to the appropriate shipping pallet immediately upon its arrival, without delay. Only after finalizing an order can operators proceed to the next one.

When loading the shipping pallets, operators must adhere to the previously mentioned constraints. Additionally, the two types of shipping pallets that the company handles are tailored to the specific dimensions of different products. This further complicates the loading operations since not every product can be loaded onto every type of pallet. The selection of a type affects not only the feasibility and efficiency of the loading process but also the information provided to customers for transportation planning purposes.

Solving this DPLP variant is particularly challenging. Even when a feasible loading plan is found, it may still be impractical for operators to implement the corresponding solution due to the unpredictability of day-to-day operations and the potential presence of packing movements that are difficult to execute in practice. Additionally, the company must inform customers, at the time of purchase, about the number and type of shipping pallets required for their order. Therefore, the company is currently interested in having a fast and accurate estimator for these values, rather than relying solely on an optimization procedure that, while correct, may require significant computation time.

5.4 Proposed Approach

The proposed approach is an extension of our previous study presented in Taccini et al. (2025). That work did not account for the different types of shipping pallets the company uses to supply its customers' orders. The company is indeed interested in a more precise prediction, not only obtaining a total expected number of pallets but also accounting for their sizes (i.e., how many pallets are needed for each available size). To this end, we developed a new approach that utilizes independent regressors and regressor chain strategies to achieve multiple outputs.

5.4.1 Single-Output Regressor Model

The approach developed in Taccini et al. (2025) involved a selection phase in which four different machine learning (ML) models were trained and compared on both the mean squared error (MSE) and the mean absolute error (MAE). At the end of this selection phase, XGBoost (Chen and Guestrin, 2016b) was identified as the most effective model. Then, the XGBoost model was enhanced through a feature optimization process. This was performed by analyzing the structure of the input dataset. At the end of this second phase, the training dataset was improved by adding more general and order-based features, such as the total number of boxes in the order, total weight and volume of the order, and average weight and volume of boxes in the order, which reduced the MSE and MAE values by 66% and 42%, respectively.

To further improve the XGBoost model, three additional optimization-based features were introduced, namely the results obtained with heuristics HS_1, HS_2, and HS_3 (Taccini et al., 2025). The first feature, HS_1, computed as the maximum between total weight divided by max pallet weight, and total volume divided by max pallet volume, can be interpreted as a lower bound (LB) for the problem. On the other hand, the second feature, HS_2, represents an upper bound (UB) for the DPLP instance, assuming that all pallets are composed exclusively of homogeneous boxes. The third added feature, HS_3, is based on a greedy heuristic that loads pallets by stacking layers of boxes with identical dimensions, starting with those having the largest area and continuing until all constraints are respected.

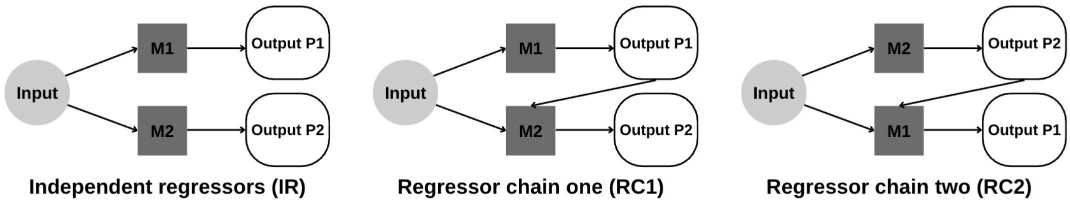
The inclusion of all these features resulted in a dataset structure, called S_3, which further decreased the MSE and MAE errors. The approach developed in Taccini et al. (2025) achieved superior performance, yielding better predictions within shorter computation times.

5.4.2 Multi-Output Regressor Model

Building on the approach in Taccini et al. (2025), we extend it to a multi-output regression model. We consider the following strategies to comply with the company's current objectives:

1. In the first scenario, we use two independent single-output regressor models, namely M_1 and M_2. Each one consists of the XGBoost model and was trained separately to predict each type of available shipping pallet, namely, pallet type $j = 1$ and pallet type

FIGURE 5.1: The three strategies developed to predict the solution of the DPLP variant under investigation.



$j = 2$. The training process assumed the dataset structure S_3. Despite its straightforward implementation, this strategy does not account for potential correlations between outputs for mixed orders, i.e., orders for which both types of shipping pallets, $j = 1$ and $j = 2$, may be present. The number of orders in this situation is around 30% of the total. We refer to this model as *independent regressors* (IR).

2. The second scenario involves training the models sequentially, with each one receiving as input both the original features, i.e., S_3, and the predictions, which we denote as P, from one of the previous models. This strategy has allowed the modeling of interdependencies between the types of pallets. We considered two strategies, since the final results/predictions are influenced by the order in which a model is applied to predict a given type of pallet. The first strategy involves using the first model, M_1, with input S_3, resulting in the first output, P_1, for the shipping pallet of type $j = 1$. Next, the second model, M_2, has as input $S_3 \cup P_1$, for the shipping pallet of type $j = 2$. We refer to this strategy as *regressor chain one* (RC1). The second strategy involves starting with the second model, M_2, which takes S_3 as input to predict the pallet of type $j = 2$. Next, we use the first model, M_1 with the input $S_3 \cup P_2$, for predicting the shipping pallet of type $j = 1$. This is named *regressor chain two* (RC2).

Figure 5.1 illustrates each of the three strategies. We consider *input* $= S_3$, while M_1 and M_2 are the XGBoost models for each type of pallet, $j = 1$ and $j = 2$, respectively. It is worth noting that several strategies were evaluated to improve the accuracy of the proposed approach, including the adoption of alternative machine learning models such as LightGBM (LGBM) (LightGBM, 2024), Random Forest, and Support Vector Machine (SVM) from scikit-learn (Pedregosa et al., 2011). However, the results presented here correspond to the configurations that achieved the best overall performance.

The tuning of the hyperparameters and the training phase for the three strategies were carried out according to the procedure described in Taccini et al. (2025). The orders were randomly split, with 80% allocated to the training phase, using a dataset containing approximately 166,000 orders from the years 2022 and 2023.

Table 5.1 reports the MSE and MAE values calculated for the test dataset, which consists of 20% of the total initial dataset. The table compares the three different strategies, IR, RC1, and RC2, based on their prediction performance. For the test dataset, while all strategies behave similarly, IR slightly outperforms the others globally and particularly for the pallet of type $j = 1$. However, for $j = 2$, we observe RC1 and RC2 show advantages depending on which measure of error we would like to prioritize, MSE or MAE.

5.5 Computational Results and Discussions

The proposed approaches were implemented in Python 3.10. We performed computational experiments on a computer equipped with an Intel Core i5-1135G7 2.40 GHz CPU, 8 GB of

TABLE 5.1: MSE and MAE values for each strategy, calculated on the test dataset. The first two columns report the average values across the two pallet types $j = 1$ and $j = 2$, while the remaining columns show the type-specific metrics.

Strategy	MSE	MAE	MSE $j = 1$	MAE $j = 1$	MSE $j = 2$	MAE $j = 2$
IR	0.659	0.256	1.133	0.407	0.185	0.105
RC1	0.663	0.257	1.129	0.414	0.198	0.100
RC2	0.660	0.267	1.140	0.430	0.181	0.104

RAM, and Linux Mint 21 Cinnamon as the operating system. The experiments utilized 30 real instances from the company, which were randomly selected from all the heterogeneous orders received in 2024 and represent different levels of heterogeneity. These orders were not considered during the training phase to guarantee a fair comparison of the strategies.

The instances are ordered by their heterogeneity, i.e., the number m of different types of boxes the order has, with a maximum of $m = 6$. The three strategies are compared with PackVol (PackVol, 2024), the software already in use by the company. It is configured to reflect the specific operational constraints of the DPLP variant under investigation; however, it does not account for the uncertainties reported in Section 5.3. We also compare our solutions with the company's expert operator solutions, i.e., the shipping pallets the operators load to fulfill each order. These values are used as a baseline for comparing all the approaches.

Table 5.2 has the results obtained for all the 30 instances (i.e., the number of pallets $j = 1$ and $j = 2$ required by each instance). In contrast, Table 5.3 reports the MAE and MSE values for each approach, with the company's operator solutions serving as the baseline for comparison. Firstly, we observe that the difference in errors between the independent regressor and the chained ones is very small, suggesting that the correlation between the two types of pallets j is weak. This could indicate that a simpler approach, such as IR, without inter-dependencies between the two types, is already able to return satisfactory predictions.

Concerning Table 5.2, as the heterogeneity of the boxes increases (ranging from 2 to 6 in instances 1–30), values for all approaches tend to increase slightly, suggesting that the problem becomes more complex as the orders' heterogeneity grows. PackVol shows a tendency to overestimate the number of pallets of type $j = 2$, which are typically associated with larger sizes. Additionally, IR, RC1, and RC2 often perform as well as, or better than, PackVol's solutions. RC2 often produces very competitive results, particularly in instances with more heterogeneity (instances 20–30).

Considering the type of pallet $j = 2$, we notice that the predictions better approximate the company's solutions. It is worth noting that PackVol gives poor results for pallets of type $j = 1$ compared to the other approaches.

Regarding the MAE and MSE values reported in Table 5.3, we conclude that RC2 is the most balanced and robust strategy, achieving the lowest overall errors and better handling the types of pallets $j = 1$. It achieves the lowest overall MSE (0.533) and lowest MAE (0.467), indicating the highest prediction accuracy globally. On the other hand, RC1 has the best results specifically for the type $j = 2$. IR remains a simple and interesting strategy, as its predictions are more accurate than those of PackVol. Notice that PackVol performs much worse than the others, especially for $j = 1$, with an MSE of 3.100 and MAE of 1.300.

5.6 Concluding Remarks

This study addressed a real-world variant of the distributor's pallet loading problem faced by a major ceramic tile manufacturer in Italy. The complexity of the problem lies not only

TABLE 5.2: Results of the 30 real-world instances for each approach. For each instance, the number of shipping pallets is reported for each type $j = 1$ and $j = 2$. The values are shown for the company's operator solutions, Packvol, and the proposed strategies: IR, RC1, and RC2.

Instance	m	Company		PackVol		IR		RC1		RC2	
		$j=1$	$j=2$	$j=1$	$j=2$	$j=1$	$j=2$	$j=1$	$j=2$	$j=1$	$j=2$
1	2	2	0	1	0	1	0	1	0	1	0
2	2	5	1	6	0	6	1	5	1	6	1
3	2	3	1	3	1	3	1	3	1	3	1
4	2	1	0	1	0	1	0	1	0	1	0
5	2	2	4	4	2	2	4	1	4	2	4
6	2	1	1	1	1	1	1	1	2	2	1
7	3	17	5	17	5	18	5	18	5	19	5
8	3	5	0	5	0	5	0	5	0	5	0
9	3	2	3	0	3	2	3	2	3	2	3
10	3	8	0	8	0	8	0	9	0	9	0
11	3	5	5	5	5	5	4	5	4	6	4
12	3	1	6	0	6	1	6	1	6	1	7
13	4	5	0	4	0	5	0	5	0	6	0
14	4	5	3	1	5	3	2	3	3	3	3
15	4	7	2	6	2	6	2	7	2	7	2
16	4	3	4	2	5	3	4	3	4	3	4
17	4	15	0	14	1	15	1	16	0	15	0
18	4	5	1	4	2	6	1	5	1	6	1
19	5	19	7	17	8	18	8	18	7	18	8
20	5	6	3	4	4	6	3	7	3	7	3
21	5	22	6	21	6	18	7	18	7	22	7
22	5	1	1	2	2	2	1	2	1	2	1
23	5	12	1	13	0	14	0	16	0	13	1
24	5	10	0	6	0	9	0	10	0	9	0
25	6	20	2	16	2	19	1	19	1	19	1
26	6	8	2	5	3	9	2	9	2	9	2
27	6	14	8	13	8	14	7	12	8	15	7
28	6	14	7	12	7	12	6	10	7	13	6
29	6	9	1	9	1	9	1	9	1	10	1
30	6	14	2	12	2	13	2	14	2	14	2

TABLE 5.3: MSE and MAE values for each approach in the 30 instances.

Model	MSE	MAE	MSE $j=1$	MAE $j=1$	MSE $j=2$	MAE $j=2$
PackVol	1.833	0.867	3.100	1.300	0.567	0.433
IR	0.800	0.500	1.300	0.700	0.300	0.300
RC1	1.183	0.517	2.200	0.867	0.167	0.167
RC2	0.533	0.467	0.833	0.700	0.233	0.233

in the geometric and operational constraints inherent to the palletization process, but also in the uncertainties associated with warehouse logistics and day-to-day operations. In this context, providing customers with accurate predictions regarding the number and types of pallets required for their orders is a critical component of efficient logistics management.

To tackle this challenge, we proposed a machine learning-based approach that integrates independent and sequential multi-output regression strategies with features derived from both historical data and optimization-based heuristics. Three prediction strategies were developed: independent regressors (IR), regressor chain one (RC1), and regressor chain two (RC2).

Computational experiments conducted on 30 real instances from the company demonstrated the effectiveness of the proposed approach. In particular, the RC2 strategy consistently achieved the best performance, obtaining the lowest overall mean squared error and mean absolute error among all tested approaches. While RC1 provided slightly better results for pallets of type $j = 2$, RC2 offered the most balanced and robust performance across different order heterogeneity. Furthermore, the IR strategy also proved to be a simple yet effective alternative, outperforming the specialized PackVol software, particularly for pallets of type $j = 1$.

Overall, the results confirm that integrating machine learning and optimization-based features can significantly enhance pallet prediction accuracy, offering practical advantages for real-world industrial logistics. The proposed approach provides an effective and scalable solution for dynamic and highly constrained environments. Future research directions include the exploration of more advanced machine learning models, such as deep learning architectures, incorporating real-time warehouse operational data to further improve prediction accuracy, and extending the approach to support dynamic re-optimization as new information becomes available during the loading process.

References

ACIMAC Research Department (Oct. 2023). *World Production and Consumption of Ceramic Tiles*. Tech. rep. 11. Manufacturing Economics Study.

Birgin, E. G., R. D. Lobato, and R. Morabito (2012). "Generating unconstrained two-dimensional non-guillotine cutting patterns by a Recursive Partitioning Algorithm". In: *Journal of the Operational Research Society* 63.2, pp. 183–200.

Bischoff, E.E., F. Janetz, and M.S.W. Ratcliff (1995). "Loading pallets with non-identical items". In: *European Journal of Operational Research* 84.3, pp. 681–692.

Bortfeldt, Andreas and Gerhard Wäscher (2013). "Constraints in container loading - A state-of-the-art review". In: *European Journal of Operational Research* 229.1, pp. 1–20. ISSN: 0377-2217.

Calzavara, Gabriele et al. (2021). "Mathematical models and heuristic algorithms for pallet building problems with practical constraints". In: *Annals of Operations Research*. ISSN: 1572-9338.

Chen, Tianqi and Carlos Guestrin (2016b). "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.

Gzara, Fatma, Samir Elhedhli, and Burak C. Yildiz (2020). "The Pallet Loading Problem: Three-dimensional bin packing with practical constraints". In: *European Journal of Operational Research* 287.3, pp. 1062–1074.

Hadj Salem, Khadija, Elsa Silva, and José Fernando Oliveira (2023). "Cutting and packing problems under uncertainty: literature review and classification framework". In: *International Transactions in Operational Research* 30.6, pp. 3329–3360. DOI: 10.1111/itor.13318.

Hodgson, Thom J. (1982). "A Combined Approach to the Pallet Loading Problem". In: *A I I E Transactions* 14.3, pp. 175–182.

Iori, Manuel et al. (2021b). "Exact solution techniques for two-dimensional cutting and packing". In: *European Journal of Operational Research* 289.2, pp. 399–415.

Layeb, Safa Bhar and Oussema Omri (2024). "Solving the Pallet Loading Problem with Deep Reinforcement Learning". In: *Handbook of Formal Optimization*. Ed. by Anand J. Kulkarni and Amir H. Gandomi. Singapore: Springer Nature Singapore, pp. 807–825.

LightGBM (2024). *Light Gradient Boosting Machine*. https://github.com/microsoft/LightGBM. Accessed: 2024-12-10.

Magnani, Matteo et al. (2025). "A Machine Learning Approach to Speed up the Solution of the Distributor's Pallet Loading Problem". In: *Numerical Computations: Theory and Algorithms*. Ed. by Yaroslav D. Sergeyev, Dmitri E. Kvasov, and Annabella Astorino. Cham: Springer Nature Switzerland, pp. 108–120. ISBN: 978-3-031-81241-5.

Nascimento, Oliviana Xavier, Thiago Alves Queiroz, and Leonardo Junqueira (2021). "Practical constraints in the container loading problem: Comprehensive formulations and exact algorithm". In: *Computers & Operations Research* 128, p. 105186. ISSN: 0305-0548.

PackVol (2024). *Optimization Software for Load Planning*. https://www.packvol.com/index.html. Accessed: 2024-12-10.

Pedregosa, Fabian et al. (2011). "Scikit-learn: Machine learning in Python". In: *the Journal of machine Learning research* 12, pp. 2825–2830.

Queiroz, T. A., F. K. Miyazawa, and Y. Wakabayashi (June 2015). "On the L-approach for generating unconstrained two-dimensional non-guillotine cutting patterns". In: *4OR* 13.2, pp. 199–219.

Silva, Elsa, José F Oliveira, and Gerhard Wäscher (2016). "The pallet loading problem: a review of solution methods and computational experiments". In: *International Transactions in Operational Research* 23.1-2, pp. 147–172.

Taccini, Marco et al. (2025). "A Hybrid Approach for Pallet Loading in Ceramic Tile Industry". In: *International Conference on Optimization and Learning*. Springer, pp. 329–340.

Wäscher, Gerhard, Heike Haußner, and Holger Schumann (2007). "An improved typology of cutting and packing problems". In: *European Journal of Operational Research* 183.3, pp. 1109–1130. ISSN: 0377-2217.

Yao, Shaowen et al. (2025). "The semi-online robotic pallet loading problem". In: *Computers & Operations Research* 174, p. 106889.

Chapter 6

Conclusions

This doctoral thesis has addressed several challenges in intralogistics optimization within the ceramic tile industry, applying data-driven OR methodologies to real-world industrial contexts. The research developed and integrated simulation, optimization, and machine learning techniques to improve the efficiency, flexibility, and reliability of logistics processes in a sector characterized by high production volumes, product heterogeneity, and complex operational constraints.

In Chapter 2, DES was applied to study and optimize the management of buffer storage areas. The analysis demonstrated how simulation can effectively model the dynamics of heterogeneous pallet flows, allowing the assessment of alternative allocation and classification strategies before their implementation. The results provided valuable insights into the trade-offs between storage capacity, operators' workload, and warehouse efficiency, confirming the role of simulation as a decision-support tool for industrial logistics planning.

Chapter 3 focused on the development of an automated DSS for optimizing transportation activities across peripheral warehouses. By combining digitalization with mathematical optimization, the proposed system minimized transportation costs and improved coordination among multiple peripheral warehouses. The implementation in a real industrial setting showed measurable benefits, including reduced operational costs, and simplified order management. This contribution demonstrated how operations research models can be effectively embedded in practical tools that directly support daily logistics operations.

In Chapter 4, a hybrid approach was proposed to estimate the number of pallets required for shipping orders of ceramic tiles. The method combined machine learning with optimization-based features, enabling fast and accurate predictions that assist decision-making in real-time e-commerce and warehouse management contexts. The approach effectively addressed a company-critical problem, providing a balance between computational efficiency and predictive accuracy.

Chapter 5 extended this hybrid approach by introducing a multi-output regression model to simultaneously predict the quantities of different pallet types. This extension further aligned the predictive model with the company needs, as pallet size selection significantly influences shipping cost. Result showed that the multi-output approach, particularly the regressor chain variant, is capable of capturing dependencies between pallet categories, thus improving the accuracy and practical benefits of the predictive system.

From an industrial perspective, all the research activities presented in this thesis were conducted in close collaboration with a leading international ceramic tile company headquartered in the ceramic district of Sassuolo. This collaboration allowed for the validation of theoretical models through real-world applications, bridging the gap between academic research and industrial practice. The developed methodologies have been implemented within the company logistics processes, contributing to measurable efficiency improvements and demonstrating the potential of operations research to drive innovation in manufacturing logistics.

With respect to deployment and operational usage, the tools developed in this thesis exhibit different levels of industrial maturity. The simulation and optimization-based decision support systems presented in Chapters 2 and 3 have been deployed within the company logistics processes and are currently used to support daily planning activities. In practice, their performance is consistent with the results reported in this thesis, leading to tangible reductions in operational costs and planning effort.

The predictive models introduced in Chapters 4 and 5 have been integrated into the company information systems as decision-support components. Although they already fulfill their intended function and have enabled the company to provide its clients with improved service and more effective data sharing for shipment planning, further refinements are currently under investigation to enhance robustness under highly dynamic conditions. In all cases, the proposed tools are designed to support human operators rather than fully replace them, automating repetitive and computationally intensive tasks while leaving exception handling and strategic decisions to human expertise.

From a scientific standpoint, the thesis contributes to the growing body of knowledge on the application of operations research and data-driven methods to intralogistics systems. It offers methodological advances in simulation-based decision support, cost-optimization modeling, and hybrid frameworks for logistics prediction. The combination of these approaches highlights the importance of integrating analytical rigor with data-driven adaptability to manage complex, dynamic, and resource-constrained environments.

Looking ahead, several research directions emerge from this work. Future studies could explore the use of stochastic and robust optimization techniques to better handle uncertainty in logistics operations, or the integration of real-time data and reinforcement learning to support adaptive decision-making in dynamic warehouse environments. Additionally, extending the hybrid predictive models to include multi-objective criteria such as balancing cost, time, and environmental impact would further improve their industrial applicability.

In conclusion, this thesis has demonstrated how the synergy between operations research, simulation, and machine learning can significantly improve intralogistics performance in the ceramic tile industry. By combining methodological innovation with practical implementation, the research contributes both to the academic advancement of logistics optimization and to the competitiveness of industrial operations in a key Italian manufacturing sector.

Appendices

Appendix A

List of Acronyms

AGV Automated Guided Vehicle.

DES Discrete Event Simulation.

DPLP Distributor Pallet Loading Problem.

DSS Decision Support System.

ERP Enterprise Resource Planning.

ILP Integer Linear Programming.

IR Independent Regressors.

LGBM Light Gradient Boosting Machine.

MAE Mean Absolute Error.

ML Machine Learning.

MOR Multi-Output Regression.

MSE Mean Squared Error.

OR Operations Research.

RC Regressor Chain.

SVM Support Vector Machine.