

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



Dynamic and optimized Picking by Line (PBL) of goods in a Distribution Centre

Alexandre Miguel Cabaça Marques de Queirós

Mestrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Jorge Pinho de Sousa

July 28, 2023

Resumo

O "picking" de pedidos é uma das operações mais críticas de armazéns e centros de distribuição. A preparação das operações de "picking", das quais depende a eficiência do armazém, pode ser influenciada por vários fatores, desde a disposição do armazém, as políticas de armazenamento, a maneira específica de gerir ordens ou até o roteamento usado por trabalhadores, fatores que se podem afetar mutuamente. Esta dissertação visa desenvolver uma ferramenta de suporte de decisão para melhorar as operações dia-a-dia de um centro de distribuição *Picking by Line* sobre condições de procura com alta variabilidade. O método *Picking by Line* é típico para produtos com elevado nível de serviço, onde um trabalhador transporta paletes de produtos para distribuir por zonas, de acordo com os pedidos respetivos de lojas. A ferramenta desenvolvida deve determinar a atribuição de lojas a zonas de maneira a melhorar as operações, minimizando a distância total percorrida no armazém. Para este propósito, foram usadas programação matemática e técnicas de optimização, para modelar as operações do armazém e poder comparar diferentes tipos de roteamento e condições, com o objetivo de melhorar a eficiência geral do centro.

Abstract

Order picking is a critical component of warehouses and distribution centers. The preparation of picking operations, in which the efficiency of a warehouse relies, can be impacted by multiple factors, from the layout of the warehouse, the storage policies, the specific way orders are handled or even the routing taken by operators, which can all affect each other. This dissertation aims to develop a decision support tool to improve the day-to-day operations for a *Picking by Line* distribution center under highly variable demand conditions. The *Picking by Line* methodology is typical for products with fast turn-over, where an operator transports pallets of products to distribute through zones, according to the matching store orders. The developed tool should determine store to zone allocation in a way that enhances the operations, minimizing the total distance travelled in the warehouse. For this purpose, mathematical programming and optimization techniques were used, to model the warehouse environment and be able to compare different routing types and conditions, in order to learn about how to improve the overall efficiency of the center.

Agradecimentos

Primeiramente, agradeço à Faculdade de Engenharia da Universidade do Porto e ao meu orientador Professor Doutor Jorge Pinho de Sousa pela oportunidade e por todo o apoio na realização desta dissertação.

Quero agradecer também à Engenheira Catarina Marques e ao Engenheiro Romão Santos por acompanharem e orientarem o meu trabalho, assim como ao INESC-TEC, em particular ao CESE, pelo ambiente acolhedor.

Agradeço ainda à minha família, amigos e colegas.

Alexandre Queirós

Contents

1	Introduction	1
1.1	Context	1
1.2	Motivation	2
1.3	Objectives	2
1.4	Methodological Approach	3
1.5	Structure of the Document	3
2	State-of-the-art	5
2.1	Background	5
2.1.1	Layout Design	5
2.1.2	Storage Policies	6
2.1.3	Order Picking Policies	7
2.1.4	Routing Policies	7
2.1.5	Picking Techniques	9
2.1.6	Information Systems	9
2.2	Brief Literature Review	10
3	Case study	13
3.1	Layout Design	13
3.2	Order Picking Policies	15
3.3	Routing Policies	15
3.4	Main Challenges	17
4	Methodological Approach	19
4.1	Adopted Approach	19
4.2	Model 1 (S-Shape routing)	20
4.2.1	Model Concepts	20
4.2.2	Mathematical formulation	22
4.3	Model 2 (Return routing)	25
4.3.1	Model Concepts	25
4.3.2	Mathematical formulation	27
4.4	Model Extension (More Accurate Demand)	29
4.4.1	Mathematical Formulation	29
5	Instance designing and processing	31
5.1	Demand Assessments	31
5.2	Distance Measurements	32
5.3	Quantity Related Assessments	33

5.4	Instance processing	34
6	Result analysis and discussion	37
6.1	Base Instances	37
6.1.1	S-Shape routing	37
6.1.2	Return routing	39
6.2	Case Study Instances	41
6.2.1	Assumptions and Simplifications	41
6.2.2	Shipping Effect	42
6.2.3	Cluster Analysis	44
6.2.4	Aisle Gap Changes	48
6.2.5	Routing Analysis	50
7	Conclusion	55
	References	57

List of Figures

2.1	Routing policies [1]	8
3.1	Layout of the case study warehouse	14
3.2	Routing in the case study warehouse	16
4.1	Routing and clusters in model 1	22
4.2	Routing and clusters in model 2	26
5.1	Layout using the given zone coordinates	32
5.2	Layout with distance measuring	33
5.3	Standard European Pallet Sizes [2]	34
5.4	Parameter Excel Sheet	35
5.5	Data Flow (Software Pipeline)	35
5.6	Example result representation	36
6.1	Model 1 picking route map	37
6.2	Model 1 result representation without extension	38
6.3	Model 1 result representation with extension	38
6.4	Extension comparisons for Model 1	38
6.5	Model 2 picking route map	39
6.6	Model 2 result representation without extension	39
6.7	Model 2 result representation with extension	40
6.8	Extension comparisons for Model 2	40
6.9	Relaxed version comparisons for Model 1	42
6.10	Model 1 picking route map	42
6.11	Result representation without shipping	43
6.12	Result representation with shipping	43
6.13	Total distance comparison with and without shipping	44
6.14	Model 1 picking route map using one cluster	44
6.15	Result representation using one cluster	45
6.16	Model 1 picking route map using two clusters	45
6.17	Result representation using two clusters	46
6.18	Model 1 picking route map using fifteen clusters	46
6.19	Result representation using fifteen clusters	47
6.20	Total distance comparison with different cluster divisions	47
6.21	Result representation using the current gap	48
6.22	Result representation using no gap	48
6.23	Result representation moving the gap up	49
6.24	Result representation moving the gap down	49

6.25	Total distance comparison with different layout gaps	50
6.26	Model 1 picking route map	50
6.27	Model 1 result representation	51
6.28	Model 2 picking route map	51
6.29	Model 2 result representation	52
6.30	Total distance comparison with different routing methods	52

Abreviaturas e Símbolos

PBL	Picking by Line
PBS	Picking by Stock
WIP	Work in Progress
MILP	Mixed Integer Linear Programming
$\mathbb{Z}$	Set of Integer Numbers

Chapter 1

Introduction

1.1 Context

Supply chains are the network of everything involved in the making and delivery of a product to a customer. They are an essential infrastructure in our current world, for example in providing essential items like food to supermarkets. New technology developments helped connect the world, by making shipping and delivering methods faster thus extending their reaching capabilities, and so the modern supply chains can be bigger and extend to larger parts of the globe than ones of the past, using economy of scales to their advantage in an attempt to grow profits.

Warehouses and distribution centers are an essential piece of supply chains, allowing the ability to store and buffer items, in order to keep stocks available and reachable to the whole world. “According to ELA/AT Kearney (2004), warehousing contributed to about 20% of the surveyed companies’ logistics costs in 2003” [1]. The goal of warehouses is to store items and then process them in a timely manner, while distribution centers add the goal of preparing and readying items for shipping, to be deployed to other locations. To achieve this, good preparation and planning are key, as mistakes resulting from an unorganized approach can lead to chain events that become complicated to untangle.

A lot of operations happen inside a warehouse: receiving, storing, in some cases replenishing or moving, sorting and packing, but among them order picking – “the process of retrieving products from storage (or buffer areas) in response to a specific customer request” [1] – is the costliest in both time and money. Order picking is a simple task to execute, but the same cannot be said about the procedures used for planning it, since many factors influence its efficiency, and such thing as a perfect solution might not exist or be feasible. Even if an optimal methodology is reached for a specific case, it is almost impossible to share it in this field of work, since every warehouse differs not only in size but in goals and demand as well. The methods used for picking are also interdependent with the methods or policies used to store, move and sort items, and even with the space and layout of the warehouse. So, decisions made about the layout design for the warehouse, storage policies, order picking policies, and routing policies for the picking can all have an effect on the efficiency of the operations.

1.2 Motivation

With the increased complexity and dimension of the supply chains, managing its operations also became harder with the need of organizing to save up costs and meet due dates. This is aggravated by modern warehouses having to supply many stores from different companies, under highly dynamic environments, complicating the logistics used while trying to keep fast response times. In the recent years, world events like the pandemic brought new conditions to the market, and the already existing tools to manage warehouse systems are not flexible enough to handle the current high levels of uncertainty.

This dissertation is associated with the daily stock replenishing of a network of proximity supermarkets that has this exact problem. A highly flexible tool that works in almost-real-time for planning the logistic and picking operations is needed, to enhance profits and the operations' efficiency, without the inherent cost of changing layouts or hiring more workers. When compared to using fixed picking techniques, a new, more dynamic approach can lower the makespan of operations, optimize space usage and maximize the overall throughput of the warehouse. These improvements can make a company more viable and increase profits, with minimal investment.

1.3 Objectives

The main goal of this proposal is to design and test a model that optimizes day-by-day operations at a distribution center, by determining the allocation of stores to picking layouts, considering product demand and internal logistics.

The performance of a distribution center can be assessed through different criteria, such as: throughput, or other ways of measuring dispatch over time; distance travelled, that has a high correlation with time; and the precision and accuracy of the service, meaning if the dates of dispatch are met and if the orders are correctly dispatched. Other objectives can be set, like space usage or the ability of having more operators working simultaneously. For this dissertation lowering the distance travelled in the center was set as the main objective, since it represents the most straightforward goal to measure, calculate and possibly improve. Although it is known that this goal does not in fact accurately represent every aspect in which the operations should be analysed, it is one of the factors that contributes most to inefficiencies in the operations. Even if, due to the occurrence of simultaneous activities, the same distance travelled can result in different throughputs, and other factors like congestion or ease of use need to be considered.

Therefore, the aim of this project is to create an optimization model that correctly translates the conditions of the *Picking by Line* distribution center and build a decision support tool that would aid in the allocation of stores to zones that exist in the center. Some variations were also considered, mostly routing related, to bring innovation and study the impact of different details on the picking performance.

1.4 Methodological Approach

A mathematical formulation of the problem was constructed to allow the use of optimization tools to reach a satisfying solution. The models created were tested with instances based on real-world data in order to evaluate their suitability. Libraries like *Google OR-Tools* were used to solve the model and reach satisfying solutions after the constraints, parameters and decision variables were created, allowing the comparison of results with other routing types or characteristics.

One of the main difficulties encountered was to represent the problem in an accurate way through a model, being careful to not oversimplify it and miss any key details, but also allowing satisfying solutions to be reached. The relation of quantities and distances, among others, can be hard to translate and should be done in a simple way. The gathering of data regarding the case study was also be a concern, as metrics and performance indicators had to be measured or created to fit the real world case.

1.5 Structure of the Document

On the current chapter 1 a brief introduction to theme was made, contextualizing picking tasks and their importance, and the motivation to explore new programming techniques that improve them regarding underlying conditions. The objectives of the dissertation were established, with a short reference to the approach that will be used to reach them.

The next chapter 2 aims to establish some general background on the techniques and different areas that can be discussed regarding warehouses and distributions centers. Going over the literature on aspects like routing and picking policies will help identify specific language and definitions that can be useful to the work.

The following chapter 3 will explain the details of the case study associated with this dissertation allowing a better understanding of the problem at hand.

The ensuing chapter's 4 purpose is to introduce the methodology used to approach the problem. It presents the optimization models created and elucidates how they translate the aspects of the real distribution center. Some different variants and an extension about the model are also shown, relevant to the objective at hand.

Chapter 5 will enlighten some aspects of how the developed tool works, how the parameters were acquired, measured and designed and then successively implemented through different software standards, transforming the input data into a solution representation.

The chapter 6 shows the different experiences that were made with conditions and data on the models, bundled with respective result analysis .

The last chapter 7 of this dissertation will make some brief conclusions about what was achieved during the project, the main challenges tackled and point out where future work could enhance the developed tool.

Chapter 2

State-of-the-art

With the motivation of understanding the retail sector, and in particular the operations at the distribution center level of the supply chain, as well as the internal logistics of distribution warehouses, a study and summary chapter consisting on the background and state-of-the-art was done. A literally review was executed in order to understand what challenges are common in the area and what approaches are used to solve them.

2.1 Background

Order picking costs and time span are normally in a direct proportion with distance travelled, by the item or operator. So, in order to minimize the throughput time of the warehouse and make it more efficient, one of the main goals is to minimize the distance travelled, keeping in mind the specific conditions that a supply chain is currently subjected to. With the aim of introducing some of the key factors that impact performance, and condensing the work stated in [1] and [3], the main factors that influence picking operations are:

- Layout Design
- Storage Policies
- Order Picking Policies
- Routing Policies
- Picking Techniques

2.1.1 Layout Design

There are two sub-problems within layout design: the general layout of the warehouse and the layout of the picking areas themselves.

The general layout of the warehouse concerns the location and size of the various departments, the most relevant for the system being the picking system itself, the loading/unloading zone, sorting and storing zones if needed, and marshalling zones. A marshalling zone is the location where items can be prepared and packed to be shipped, so it's location and size is also fundamental to

the picking system. Other departments like offices and sanitary facilities have less impact when compared to the ones mentioned before, but their placement must also be decided.

When it comes to the picking area itself, normally aisles, and corridors that act as passages between shelves of items or goods, tend to be used. The height, length and number of aisles must be decided, but the method for picking can heavily influence this decision. Picking can be done by humans or most recently machines, in automated systems. For the automated warehouse, either conveyors and dispensers can be used or picking robots. “The majority of warehouses employ humans for order picking. Among these, the picker-to-parts systems, where the order picker walks or drives along the aisles to pick items, are most common” [1]. There is also the *parts-to-picker* method, involving more automated storage systems with cranes that retrieve items from aisles to a certain location, and the *put* systems, where items are first collected from aisles and distributed by orders second.

2.1.2 Storage Policies

Some of the methods used to store items in aisles are: *random*, *closest open location*, *dedicated*, *full turnover* and *class based storage*. Apart from those five methods, it is also relevant to note the *Forward-reserve* allocation method.

The *Forward-reserve* method splits the warehouse into two different areas. The forward area is a smaller space where items will be picked from, while the reserve area is a place to store the bulk stock of items. The smaller picking space makes the operations faster, but this method needs the added task of replenishing the forward area when its stock runs out, having to move items from the reserve area to it. The size of the areas and amount of stock to keep is a key decision in this method.

As for storing the items in the aisles, the *random* method assigns a random empty slot in the warehouse for each item received. This method shows high space utilization, but will only work in environments where computer information systems are present, to keep track of where the items are placed. In the *closest open location* method items will be stored in the first open slot found. This results in storage spaces being occupied around the loading/unloading zones and empty away from them. The *dedicated storage* maps a permanent slot to each type of item. This leads to the operators getting familiar with where items are placed, which makes picking faster, but it also wastes storage space for products that are out of stock, which in a high variability system is a massive disadvantage. *Full turnover* distributes items according to rankings done based on data. Turnover rate measures the dispatch rate of items and can be used for this goal, but there is also an index called COI (cube-per-order) that considers the space of each item as well. This can lead to shifts in item placement over time, making it harder for operators to get familiar on where to go. The last method, *class based storage*, assigns classes to each item according to a Pareto method, where a low percentage of the items represent a high percentage of the turnover.

2.1.3 Order Picking Policies

Every picking task involves either loading or unloading specific picking pallets into aisles according to orders. Order picking policies concerns how the products in said orders are handled.

Picking by Order, or *Picking by Stock*, is a method where the operator will construct an order by picking up every item from its storing place, before moving to the next. This is a simple method, since it has fast response time for single picking orders, but it might make the operator walk back and forth inside the warehouse, causing inefficiencies in travelled distance when managing a large number of orders.

Picking by Line, *PBL*, is a specific method for picking, commonly used for items that have short life span, like groceries, or items that should be dispatched in a hasty manner. In *PBL*, items received in the distribution center are firstly broken down from the received pallets, then distributed to zones laid out in a grid, each zone corresponding to a specific customer or order. More than one zone can be assigned to an order, to distinguish different type of items or simply to have more space, and items do not tend to stay stored in the zones for long before being moved out to dispatch.

The grouping of orders, named *batching*, can be done according to the time delivery window of orders or the proximity of items in storage for each order. *Batching* can save time, by letting the operator pick up items near each other without having to travel back and forth from the deposit zone each time, but it also adds the need of splitting the collected items into the specific orders later.

Zoning, diving the warehouse into different areas, can also be considered in combination with the policies. This allows many operators to work in the picking at the same time. *Zoning* can be done by *progressive assembly*, where the picking is started by one operator and then passed to the next, or by *synchronized* picking, where all the workers pick at the same time, adding then a task of sorting the orders at the end.

2.1.4 Routing Policies

Routing polices considers the individual route for each operator to pick the items in. This is similar to the Travelling Salesman Problem, where a salesman, starting from home, must visit several cities and return home. The distance between cities is known, and the goal is to determine the order of visits that minimizes the total travel distance. On a warehouse the operator acts as the salesman and the cities are the locations in the aisles where he must pick up items from, although the existence of aisles restrains the number of solutions possible.

The method of travel and pick up also concerns the routing policies, and affects every other policy. Using a vehicle to travel and a large cart to collect items in the warehouse is intrinsically different from walking and having a small basket. The size between aisles is also a concern if multiple operators work on the same route, since passage could become obstructed. It is good practice to consider what bottlenecks could happen from mistakes in the picking, since in real

warehouses not everything goes according to plan, and this is sometimes a deciding factor in picking a routing policy over another.

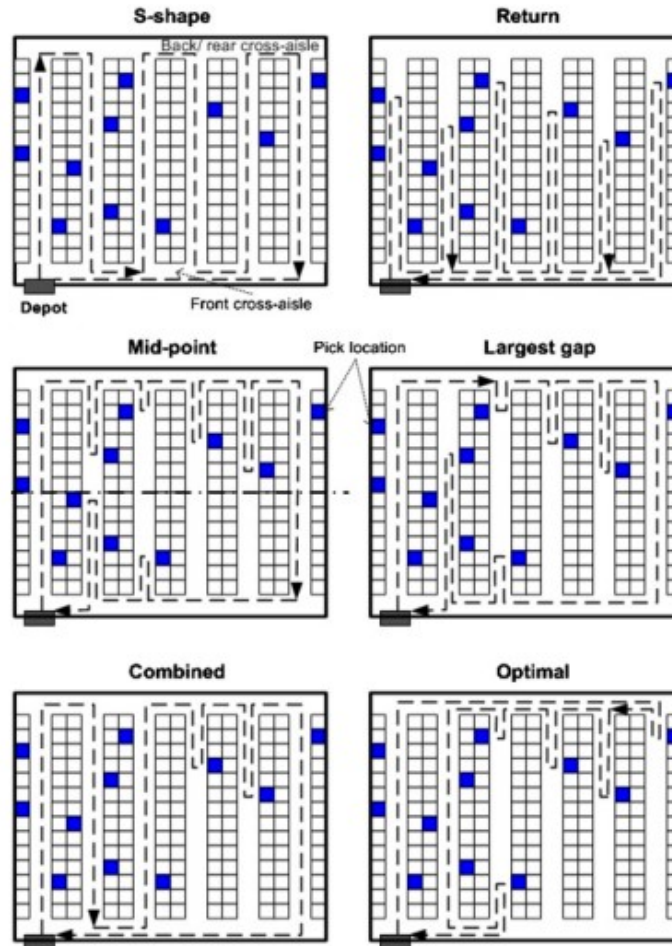


Figure 2.1: Routing policies [1]

Achieving the best route for each operator would take a great amount of time, so mostly heuristics are used to reach satisfying results in a short amount of time. Routing depends on the layout of the aisles, but some of the common heuristics used are the *S-shape*, *return*, *mid-point*, *largest gap* and *combined*, and can be seen on the figure above. *S-shape* routing consists of the operator travelling through an aisle completely then travelling the opposite way on the next aisle, while aisles without items to be picked are not entered. The *return* method has the operator travel the aisle as long as he needs to pick up the items, then returning to the start of the aisle before repeating the process on the next aisle. *Mid-point* splits the aisles in half, in upper sections and lower sections, the operator first navigates one of the sections for all aisles followed by the other. The *largest gap* acts as the *mid-point* method, but instead of splitting aisles by the middle it splits each aisle by the largest distance between either items to be picked or items and the end of the aisles. The *combined* method allows the operator to either transverse the aisle completely or only

as long as he needs, the choice being dynamically made for each aisle.

2.1.5 Picking Techniques

In order for operators to search and localize items on their picking list different techniques can be employed. This is important, since bad techniques can lead to longer searching times or even errors on the picking, either creating wasted time to fix them or resulting in wrong deliveries, lowering the warehouse efficiency.

RF(Radio Frequency) Scanners are common in warehouses because of their high efficiency. They act like terminals from which operators can receive orders, and also allow them to scan bar codes on the items, confirming the correct pick-up. This validation at picking skips having a later phase to confirm orders, and identifying mistakes early saves wasted time when compared to a final check.

Picking by Light, which is sometimes referred to as *PBL*, but is not to be mistaken with *Picking by Line*, is a method where a light turns on next to the item to be picked, sometimes even with information on the quantity to be picked. There is also a *Put by Light* technique that works the same way but for storing items instead.

Picking by Voice is another method, where operators use an earpiece to receive audio information about which items to pick. This information can be generated by other operators, but it is mainly obtained from integrated computer systems. This method allows feedback from the operator, that can request new information via audio as well. The main benefit of this technique is that it eliminates the time taken to interpretate visual signs, and also allows the operator to work while receiving new picking lists.

A new technology called *Picking by Vision* utilizes smart glasses that guide operators inside the warehouse by showing them useful symbols and arrows.

The last technique, useful for keeping track of items stored, is called *RFID*, and it utilizes radio frequency identification to locate tags. Tags can either be active and generate their own signal or passive and only allow the reading object to find them, resulting in a lower signal distance. This technology supports not only the picking but also the information system of the warehouse, if one exists.

2.1.6 Information Systems

It is critical to consider what type of information system is integrated in the warehouse or distribution center to be able to choose according techniques to use. Some of the policies are not viable without such system, for example the *random* storing method. If one operator were to randomly store an item without a way to keep track of where it was placed, another operator might have to move around the whole warehouse before finding it. It is not a main goal of this project to develop an information system, but it is important to know aspects such as capacity, used or left, orders and items received, item placement, pallet preparation and equally important how to relay or visualize such information.

2.2 Brief Literature Review

The methodologies written about in the above chapter appear over many dissertations and papers about this field of research, but each work tends to focus on specific parts of the problem, or on case studies under specific conditions. Problem definition and identification of areas to act on seem a key step to a successful work, and it also distinguishes works from one another, allowing a high diversity of approaches within this subject. The order picking field has a varied supply of literature works, and even papers that review said literature to different intents.

In [3] a method to approach layout configuration is addressed. Work [4] considers the effect of different storage policies on optimizing a retail warehouse, through an integer-programming model. The different order picking policies used by different real warehouses are catalogued and analysed according to picking volumes, picking rates and number of different items in paper [5]. [6] and [7] both review batching techniques, while [8] considers a problem with a specific number of items that should be zoned to different operators and areas, choosing picker and product locations to minimize the cycle time of order picking. As to routing policies, [9] evaluates the efficiency of the different methods, coming to concrete conclusions on the ranking for a specific warehouse case. [10] shows a complex routing problem, where different blocks of aisles exist, and vertical distances are taken into account, which complicates the distance calculations and routing choices. [11] studies the interaction between different picking, storage and routing policies. Many other works exist and could be cited, and [12] investigates literature review to determine how related the different planning problems are, coming to the conclusion that the majority of papers focus on batching, routing and storage assignments.

In a different take [13] is written about a robotic *parts-to-picker* environment, using integer-programming to analyze sequencing of orders and picking robots. [14] shares with this project the contemplation of dynamic orders and consequently dynamic picking lines, using heuristics to solve order batching. [15] analyses the literature in this field to reflect on human factors in order picking, being that labour is costly and can lead to mistakes in *picker-to-parts* warehouses. Human factors are stated as a research gap in the area, since work safety, accidents and work conditions that can lead to wrong order fulfillment are not often pondered in the literature.

A whole new section of literature can be found related to cross docking distribution centers, since they share some of the picking characteristics, but also differ in some aspects. Similar to *Picking by Line*, the focus is on dispatching items quickly and effectively, as opposed to using long lasting storage techniques. Work [16] approaches the designs of optimal box types for these warehouses. A big part of this specific problem is related with arriving and shipping trucks, since their scheduling heavily influences operation time, as stated in [17]. [18] studies a parallel topic, in truck routing with several working docks and different types of transports, some refrigerated. Finally, [19] does a literature review on cross docking, examining the occurrence of aspects such as the number of docks, transport type, warehouse shape, time items are stored for and inbound trucks over the literature.

An important paper [20] addresses keys areas that are relevant to the project, especially in

problem definition. The work focus on a cross-docking environment, akin to the one found on a *PBL* system. In both instances, items that arrive at a distribution center are laid out across a grid, consisting of zones, on the floor, according to which store the zone belongs to. “The main contribution of this paper is to propose a new warehouse layout design approach namely for the allocation to retail stores to pre-dimensioned floor spaces, based in mathematical programming techniques, that is suited for a cross-docking environment.” [20]. Due to the alignment of the problem, this paper acted as a starting point to the problem definition and methodological approach.

This work is oriented to the optimization of a dynamic *Picking by Line* distribution center, so that will be the main focus of this paper. The literature is vast on the different kind of problems and warehouses present in the area, making the task of summing up all the aspects and angles unfeasible. That being said, it seems like storing, routing and batching policies are the most explored, leaving human factors or the automation of lines as loose ends that can be explored more in depth, since they do not concern most works. Layout designs seem to have a standard definition in aisles, but this aspect could be changed, with imagination being the limiting factor.

Chapter 3

Case study

This dissertation is inspired by a case study of a distribution center responsible for a network of proximity supermarkets. With the goal of enhancing its operations' efficiency, the understanding of the scope of the problem and its clear definition is vital.

The first recognition is that the warehouse operations can be split into two separate methods. Big and irregular products are dealt with a *Picking by Stock* tactic, with objects being stored in shelves and picked by an operator when needed for an order, but these fall out of the range of this project. The other operations, which will be the focus of the project, are handled in a *Picking by Line* method. This is the case for products that need to be dispatched quickly, mostly on a day-to-day basis, with some few exceptions.

For the *PBL* approach, the distribution center receives trucks of products, and has to organize them to be expedited according to order demands from stores, that involve different quantities of products of different types. To do this, the operators distribute the received products into zones laid out on the center's floor, with each zone belonging to a single store. After the orders are fulfilled with the respective products, the products placed in the zones are moved to an expedition area.

3.1 Layout Design

Since this dissertation follows along a case study with a defined general layout, no meaningful choices are to be made to change it, with loading, unloading and other areas being already set. Therefore, the region of the warehouse that will be impacted by any developments is the layout of the picking area, shown in picture [3.1](#).

The picking layout that can be seen in [3.1](#) has two center aisles with corridors to each side, that human operators utilize to perform the picking. The aisles are divided in zones, with each zone being an according floor space that can be attributed to storing pallets of products that will eventually end up at the same store. An aisle gap exists at roughly the halfway point, which is a key feature of the layout since it provides a shortcut for different routes. Aside from the center aisles, another aisle exists to the right of the warehouse, being supplied by the right corridor. Smaller

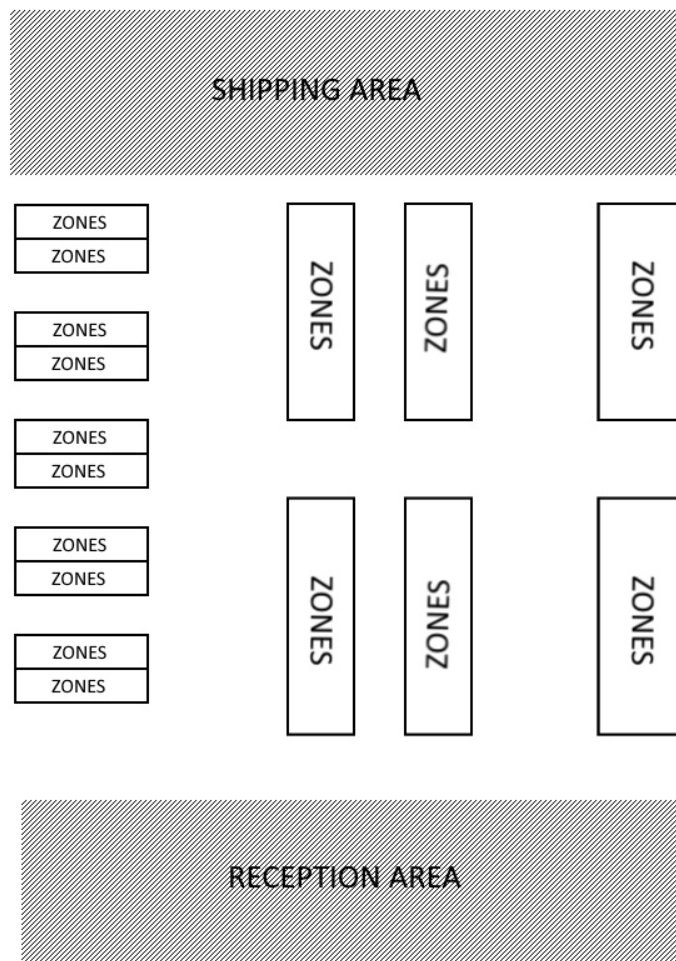


Figure 3.1: Layout of the case study warehouse

groups of zones are spread horizontally on the left of the warehouse, and these have generally a lower capacity for storing products.

The reception area is assumed to be in the lower part of the warehouse, contrary to the shipping area and according marshalling area, which are placed on the top. None of these areas are completely defined, and, instead, individual distances from each zone to the loose region are set. This represents some drawbacks, has inside the wide reception region operators still have to move around to transport product or go back to their original posts, and this is not going to be considered in the problem. The same can be stated for marshalling zones, as their loose definition does not allow the contemplation of the need for ordering outgoing shipments, simplifying the problem but also losing some of its intricate details in the process.

3.2 Order Picking Policies

Picking by Line systems act similarly to cross-docking ones, therefore *Storage Policies* can be skipped, because the picking is already intertwined with the storing.

Trucks of products arrive from suppliers to the distribution center, and the main objective is to organize them to be shipped according to order demands from stores, that involve different quantities of products of different types. To achieve this, the operators in the center distribute the received products, using specific picking pallets, into the zones on the center's floor, with each zone belonging to a single store, although more than one zone can belong to the same store if such is needed for capacity purposes. The preparation of the picking pallets is a key detail of this process, but it is considered that pallets can be prepared in a way that suits the routing process. After the zones are filled with the respective products, the products placed in them are moved to an expedition area, using shipping pallets that can differ in size from picking pallets. The operations are typically done in a day, hence the day-to-day term assigned to this dissertation, with rare exceptions of products that stay in a zone for longer.

Orders which take longer than a day to complete are denominated "WIP", as in Work In Progress. After some analysis it was decided to not include them in the problem, as the goal was to solve the attribution of stores for day-by-day, and for the results of the allocation of "WIP" orders to mean anything substantial, the proposed model would have to consider more than a day at a time, adding complexity and time to the solving process.

Picking pallets are prepared for the operators picking, with products that allow the delivery for different zones and according store's demand, minimizing the time spent by the operator walking back to the reception area, where picking pallets are prepared. It only makes sense to consider *Batching* when talking about the picking phase of operations, the distribution of the received products into the zones, and instead of grouping orders, it relates to grouping different types of product for different orders in a single picking pallet to economize distance and time further.

The number of operators on the working fleet is important to the efficiency of the operations but no extensive analysis was done on this factor.

3.3 Routing Policies

For picking routes, operators start their routing at the reception area and end it there as well. With simplicity and ease of use in mind, the routing in the warehouse is made resorting to the *S-Shape* method, keeping the traffic going in one way only. In each specific route the zones are visited sequentially, so there is no possibility of having to retreat to a zone that was already passed by, meaning there is no crossing between operators walking in different directions.

In the main center vertical aisles the routing can be represented more as an "U-shape" than an *S-Shape*, with operators going up one side of the aisle and back down in the other. There are two opportunities to cross from one side of the aisle to the other, either at the end of the aisle or at the gap that exists in each aisle. This choice heavily affects the distance travelled in the warehouse,

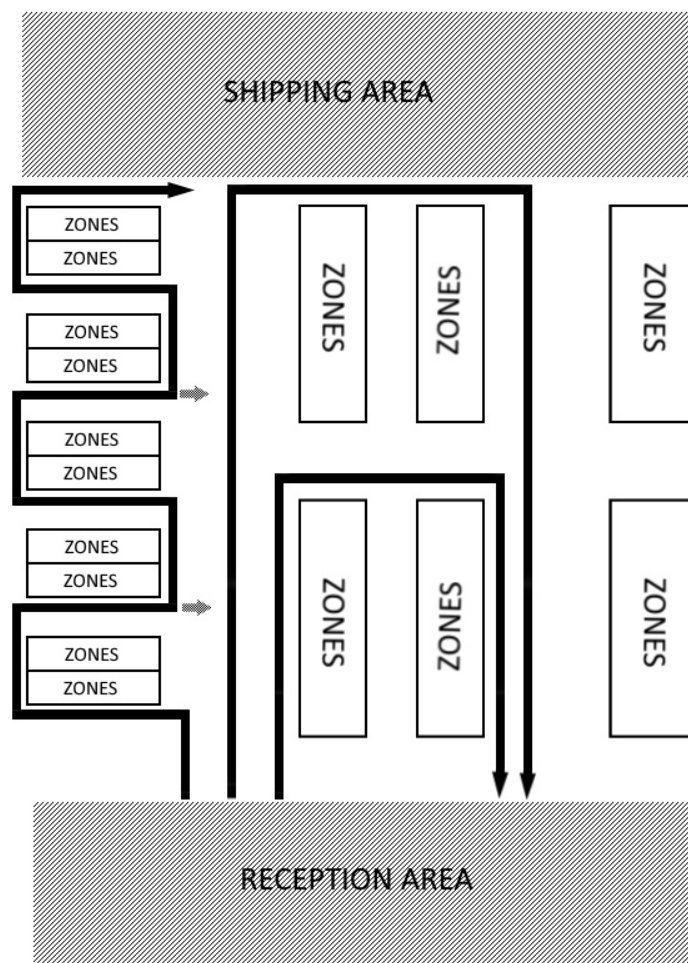


Figure 3.2: Routing in the case study warehouse

and is one of the key details of the existing routing, since its existence allows the variance in results of different solutions to allocate stores to the zones. It is worth reminding that operators supply the right zones of the warehouse when coming down the main path.

In the left side aisles, a more typical *S-Shape* routing is done, as can be seen in image 3.2. The biggest point of this routing is how it is integrated with the routing of the main aisles, as operators can only finish their routes by returning to the reception area, which, by the nature of the routing, can only be done by following the path in the center aisles to its end. There are some points along the *S-Shape* of the side aisles where the crossing to the center path can be done.

As was previously mentioned, since the marshalling layout is not defined in enough detail to consider truck load optimization, only the rough sketch of the shipping process is contemplated. This is done by considering that each shipping route starts at a product filled zone and ends at an expedition area, with this distance being known. The pathing operators do to reach the product filled zone or after delivering the shipping pallets to the expedition area is not stated or taken into account.

Operators use fork-lifts to do the routing, transporting only one pallet at a time for both picking and shipping. This also influenced the current choice for routing policies, since congestion is harder to manage when dealing with vehicles. In this case study *Picking by Voice* is used to help operators along their routing to supply zones. The next zone to visit and the amount of products to supply are communicated to guide the operations.

3.4 Main Challenges

The picking operations on the distribution center are the essential point that this dissertation aims to study and possibly improve. The specific metric used to measure solutions and the impact of different methods is the distance travelled by operators to fulfill the store orders. This metric heavily affects the efficiency of the picking activities, although it might eventually not reflect all the objectives for the warehouse, as makespan or throughput are also important, but rely on a time element which is harder to measure and evaluate and can depend on other factors such as the number of workers and the working schedule. Shipping operations also impact the warehouse greatly, and are considered to some extent, but are kept simple and this could be a point for future development.

Currently at the distribution center, stores have fixed zones allocated to them, which causes inefficiencies, considering how order demand is uncertain. Therefore, the key problem at hand is the attribution of stores to the zones in a way that enhances the operations at the center. In order to calculate how different allocation of stores impacts the distance travelled, a methodological approach was adopted.

Chapter 4

Methodological Approach

4.1 Adopted Approach

Optimization methods were used to address the problem at hand, since they provide the right tools for solving complex decisions, where the goal is to find the best course of action among many possibilities. In a general definition, optimization problems consist on finding a solution that maximizes or minimizes an objective function, from an extensive set of possible solutions.

The use of mathematical optimization is common practice to solve some problems in the industries of supply chains and manufacturing, since it helps create plans for facility locations, routing and scheduling, among others. Even though these type of problems can be well defined, they are characterized by having a high number of possible solutions, and without a precise way to find the best solution, optimization helps navigate to an appropriate one.

To reach successful solutions through optimization and mathematical programming, it is important to model the problem in a way that correctly translates the real world constraints and conditions, and for this case a *MILP*, Mixed Integer Linear Programming, type of model was chosen, since it dealt with the continuous and binary decision variables created for the problem. The linearity of the model is an important trait, as it can lead to some simplifications of the case details.

In order to solve the *MILP* model, a *Python Google OR-Tools* module was used, considering as this software supports this type of problem and formulation. *Google OR-Tools* allows the use of different algorithms to reach solutions, such as commercial solvers provided the user has the licenses, like *CPLEX*, or the chosen *SCIP*. "SCIP is currently one of the fastest non-commercial solvers for mixed integer programming (MIP) and mixed integer nonlinear programming (MINLP)." [21], so it was the adequate choice for this dissertation.

The module allows the creation of the model, through the respective decision variables, constraints, parameters and objective function, and gives feedback to the feasibility of the problem, and solution reached. It is important to be aware of the concept of the optimality gap, since the solver can not always reach an optimal solution in a desired amount of time, and this concept gives an indicator of the gap between the result achieved and an estimated best bound solution. The optimality gap is a great indicator to show if there is still room for progress, even in a state

where it might not be feasible to currently improve the solution in an acceptable amount of time with the ongoing methodology.

To answer the main challenges of the case study, and reach optimal or satisfying solutions, two different *MILP* models were formulated. The first model presents a fixed routing approach, by grouping zones into clusters, inspired by the work done in [20], with the intent of translating the current routing approach. The second model provides an alternate routing strategy, based on the *return* routing method.

4.2 Model 1 (S-Shape routing)

This formulation models the current routing done in the presented case study 3. It utilizes the same approach as [20], by dividing the zones into clusters with associated fixed picking routes. The proposed model aims to derive the allocation of stores to zones that minimizes the total distance travelled in the warehouse.

4.2.1 Model Concepts

The warehouse operations can be split into two parts:

- The picking phase, consisting of distributing the received items through the zones according to the store to zone attribution.
- The shipping phase, where it is assumed the operator will carry the product from the fulfilled zones to an expedition area.

Firstly, it is important to allocate stores to zones in a way that does not exceed the capacity of a zone, allocating more than one zone per store when needed, as is common with the [20] paper. Another consideration is that zones of the same store should be placed sequentially, to ease the shipping operations, which is achieved in a different way than the work the model was inspired on.

When a store needs more than one zone, a demand splitting factor must be taken into account for the demand of each zone, since it does not equal the total demand of the store anymore. Furthermore, since zones can have different capacities for storing products this issue turns even more complex. In ideal conditions, the demand split between zones would not be either even or proportionate to zone capacity, but it would solely depend on distance for both picking and shipping operations. Despite this, reaching this solution through linear methods is hard to conceive.

Instead, the actual solution implemented on the model is a simplification, in order to keep the model linear and faster to solve. It starts by estimating how many pallets of products each store will need as a parameter. It then calculates how much product would be placed in each pallet in an even split. But since zones can store different amounts of pallets, each zone has allocated demand equal to the calculated demand by pallet multiplied by the amount of pallets that fit in said zone. This simplification causes an issue though, as the calculated demand for each zone counts every pallet that could fit in that zone as used, when this might not translate the real usage. Because of

this rounding up of demand, the model takes into account the capacity of each zone as well as the distance when attributing it to a store, which is not ideal. The silver lining is that the model will line up store order size and zone capacity to minimize the rounding up of demand which affects the total distance travelled, basically taking into account space usage. A more accurate solution was devised in the form of an extension, in 4.4, although it demands more processing time.

It is assumed that the layout in the warehouse is close to the one shown before in section 3.1, with zones being sequenced and distributed in three aisle-like shapes, two placed in the middle, while the last is on the right, all with a gap at roughly the halfway point, and some additional groups of zones to the left side, akin to what was explained for the case study.

To help model the situation, zones were mapped out to different clusters, in this case a cluster being a group of zones close together that an operator can supply in the same route, using the same approach as [20]. Therefore, each cluster has a fixed associated route the operator will travel along for the picking. It is assumed that picking pallets are prepared for the picking activities, consisting of the products of different types to fulfill the demand of the zones.

It is important to point out that although some routes could overlap and also supply zones belonging to other clusters, this is not taken into account in the current model, keeping the clusters split when it comes to both routing and picking, although a shared picking amount could have been considered. The division of the zones into clusters can also be changed, to represent a different pathing taken or different aisle structure.

While in the picking phase, the operator will carry pallets along the *S-shape* routes, beginning and ending in the reception area, as shown in figure 4.1. The travelling is done in sequence and one way only, so the operator can't turn back to zones he already passed by, although a shortcut through the aisle gap can be used depending on which zones are being supplied. For the aisles the routing resembles a "U-shape" since the route uses both corridors, as for the side zones the *S-shape* can be better seen.

Specifically for the case study situation, the left groups of zones can be divided into more or less clusters, ending the *S-shape* early at specific points represented on the image, and returning to the reception area through a route similar to to the ones for the main aisles, either through the existing aisle gap or at the end of the warehouse. The cluster division can save distance, and since it depends on how many groups of side zones there are, the demand for stores that can be placed there and the amount an operator can carry in a picking pallet, it will be explored further when analysing results. The zones on the right are supplied by the routing that comes down the right corridor, that is common to some of the center zones.

To calculate the distance travelled on the picking phase, the number of times a route to each cluster needs to be performed is determined based on the demand of stores attributed to zones that belong to the cluster and on the amount of product that can be carried on a picking pallet. For every cluster, the number of performed routes is multiplied by the distance of the route associated with it to determine the total picking distance. This differs from the cited work [20], which determines the amount of routes based on average demand, which would not be appropriate for a case with highly uncertain demand.

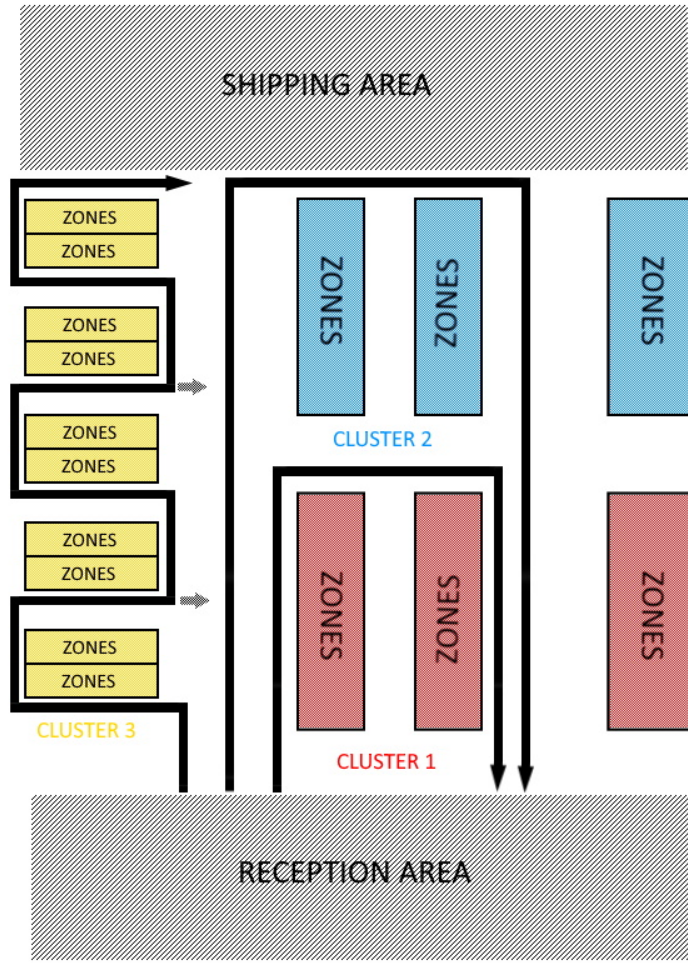


Figure 4.1: Routing and clusters in model 1

As for the shipping phase, a unique distance is set for each zone, to represent the travelling the operator will do moving a pallet from the specific zone to the expedition area. Details like the truck loading order or different marshalling zones are not considered.

Calculating the distance for the shipping phase is easier, as after determining how many shipping pallets are needed to transport the product for each zone, a multiplication by the set distance to the expedition area for that zone is done, summing to the total shipping distance.

4.2.2 Mathematical formulation

Indices

The following indices were indicated:

$i \in I = \{0, 1, 2, \dots, m\}$ - Set of Zones

$j \in J = \{0, 1, 2, \dots, n\}$ - Set of Stores

$k \in K = \{0, 1, 2, \dots, o\}$ - Set of Products

$c \in C = \{0, 1, 2, \dots, p\}$ - Set of Clusters

Decision Variables

The following decision variables were created:

$X_{ij} \in \{0, 1\}$ – 1 if zone i is allocated to store j , 0 if otherwise.

$N_c \in \mathbb{Z}$ – Number of times the picking route associated with cluster c is performed.

$NS_i \in \mathbb{Z}$ – Number of shipping pallets needed in zone i .

$E_{ij} \in \{0, 1\}$ – Decision variable that represents the end, in zone i , of a sequence of zones that belong to the same store j .

Parameters

The following parameters were set and categorized:

Demand

PS_{jk} - Demand of product k from the store j , in product units.

V_k - Factor that specifies how each product type k affects the overall quantity of a zone or pallet, akin to a weight or volume.

Distances

dP_c - Distance for the picking route associated with cluster c , in meters.

dS_i - Distance from zone i to the shipping area, in meters.

Quantities

QZ_i - Maximum quantity of product in zone i , in product units, with the weight factor taken into account.

QP - Maximum quantity of product that an operator can carry in a picking pallet, in product units, with the weight factor taken into account.

QS - Maximum quantity of product that an operator can carry in a shipping pallet, in product units, with the weight factor taken into account.

Other

$Y_{ic} \in \{0, 1\}$ - 1 if zone i belongs to cluster c , 0 if otherwise.

$S_j = \sum_{k=0}^K (PS_{jk} * V_k)$, $\forall j$ - Total capacity, in product units, needed to fulfill the demand of store j considering how every product type affects it. This parameter is auxiliary to help understand the constraints, as it could be substituted by the corresponding expression when it appears.

$Z_j \geq S_j / QP$, $\forall j$ - Number of picking pallets, in units, that are needed for store j . This is inserted as a parameter to keep the model linear, although in an ideal scenario it could be a decision variable.

Objective function

Minimize the total distance travelled:

$$\sum_{c=0}^C (N_c * dP_c) + \sum_{i=0}^I (NS_i * dS_i) - (1a)$$

The first term of expression (1a) is associated with the total picking distance, which is the sum of the number of times the picking route associated with each cluster is performed multiplied by the distance of the routing associated with the cluster. While the second term is the total shipping distance, which uses the amount of shipping pallets in each zone multiplied by the distance from the zone to the shipping area.

Constraints

$$\sum_{j=0}^J (X_{ij}) \leq 1, \forall i - (2)$$

$$\sum_{i=0}^I (X_{ij} * QZ_i) \geq S_j, \forall j - (3)$$

$$N_c \geq \sum_{i=0}^I (Y_{ic} * \sum_{j=0}^J (\frac{X_{ij} * S_j * QZ_i}{QP * QP * Z_j}), \forall c - (4a)$$

$$NS_i \geq \sum_{j=0}^J (\frac{X_{ij} * S_j * QZ_i}{QS * QP * Z_j}), \forall i - (5)$$

$$X_{ij} \leq X_{(i+1)j} + E_{ij}, \forall i < I, \forall j - (6)$$

$$X_{ij} \leq E_{ij}, \forall i=I, \forall j - (7)$$

$$\sum_{i=0}^I (E_{ij}) \leq 1, \forall j - (8)$$

Constraint (2) ensures that every zone can have a maximum of one store allocated to it and constraint (3) makes sure to allocate the necessary number of zones to each store to satisfy the total store demand.

Constraint (4a) determines the number of times the picking route associated with each cluster is performed must be enough to satisfy the demand of the zones that belong to the cluster, considering the stores attributed to those zones and how much a picking pallet can carry in each visit. A demand balance factor is taken into account, so that if a store is attributed to more than one zone, the demand is split evenly through zones, according to zone capacity, as was explained in the model concepts. A extension that describes the demand more accurately is further explained in 4.4.

Constraint (5) assures that there are enough shipping pallets in each zone for the amount of product placed there, considering which store is allocated to the zone, the total space needed for the specific store and the amount a shipping pallet can carry. Since a store can be attributed to more than one zone, the number of shipping pallets is distributed with the same demand balancing factor as in constraint (4).

Constraint (6) and (7) assure that, when more than one zone is needed for a unique store, they are sequential. Constraint (8) complements those constraints, as it guarantees that only one sequence of zones belonging to the same store exists, by limiting the zones where successive zone placing can end.

4.3 Model 2 (Return routing)

The second model was created to be able to test a variant routing, based on the *return* method explained in the background section. Most details of the problem stay the same, therefore the formulation only differs a little from the first base model, and only the changes are shown.

4.3.1 Model Concepts

The main difference of this model is that it proposes a different type of picking routing, that utilizes the *return* method, instead of the *S-shape* routing that was modeled in the chapter above and is currently utilized in the case study.

In the picking phase, routes performed still resemble an "U-shape", although instead of using both corridors to achieve this, each of the routes are fully performed on a single side of the aisle, as shown in figure 4.2. Along each route, operators start supplying the closest zone to the reception area, and move on to the next closest zone. When finished with the current picking pallet, instead of finishing the remaining route like happened beforehand, the operator will return to the reception area, moving back in the opposite way. Instead of fixed routes, these are dynamic and different route lengths happen as a consequence of the different points of turning around. Traffic can be made in both ways, but the travelling is still done in sequence, with the correct sequencing of the zones based on distance being crucial to this model.

Since routes are fully performed from one side of the aisle, both sides of the aisle are isolated from each other. A new division in clusters, different from the one stated in the above chapter was made, so that routes could be categorized according to the cluster, or side, they actually supply. A cluster can then be defined as the zones an operator can supply in a straight line from one side of

the aisle. Routes still start and end in the reception area, but at slightly different points from each other if wanted.

For the left side zones specific to this case study, different clusters were mapped out according to which side they can be supplied from. An operator will follow a branch of the route common to all, then step sideways as far into the group as he needs to satisfy demand, before returning to the common branch route and finishing it. It would save distance to calculate another *return* method for the common branch, but this would make the model more complex, which might not be necessary at this point, since the comparison between models is already hard. As for the zones in the right side, they can be supplied by the right corridor, simultaneously as the right facing middle zones.

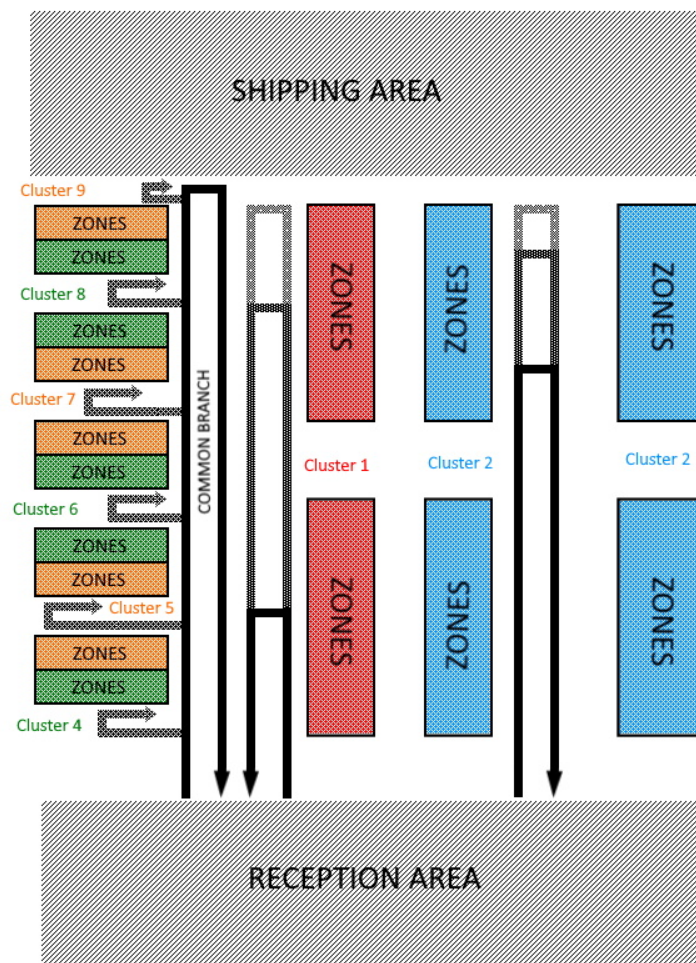


Figure 4.2: Routing and clusters in model 2

The return point of a route happens in the last zone that can be supplied in it, either because a new pallet of product is needed or because the picking operations have ended. It is useful to use the inverse order of sequencing from the picking of these zones to calculate these points, since they are the last points visited in each route. For each cluster, the first point calculated is the zone

furthest away from the reception area, since no route that returned before could have supplied that zone. From there the other points are estimated for each zone, taking into account the amount of routes that return in zones further away from the reception area, the amount of product those routes can supply, which is directly linked with the quantity carried in a picking pallet, and the allocated store demand to each zone. In a cascade, all the “return” points are estimated. It is relevant to understand that the model will try to place the points as close to the reception area as it can, since this minimizes the distance travelled in the routes, and the usage of the inverse sequencing, from the furthest away to the closest, counter balances this, making the model place the necessary points to satisfy the demand of the outermost zones.

Since, in this method, the routes are a straight line, and in sequence from closest zone to furthest away, knowing the point where the operator turns back to return to the reception area allows a simplification of the distance travelled, based on doubling the distance of the reception area to the zone where the turning around happens. To calculate the distance travelled on the picking phase, the amount of routes that return to the start in each zone is determined, which allows the estimation of the total picking distance.

For the specific scenario of the case study, because of the need for pathing to the left groups of zones to return through a route in the main center space, an additional distance is calculated. The distance associated with the common branch used for the side zones is then added, by calculating the number of times this route is used.

The shipping, other operations and remaining aspects of the model stay the same.

4.3.2 Mathematical formulation

Decision Variables

The following new decision variables were created:

NR_i – Number of routes that return in zone i , meaning that zone i is the last zone supplied in the picking route before the operators turn around to end the routes.

NB – Number of times the route in the common branch is needed to supply all the zones affected by this case oriented detail is performed.

Parameters

The following new parameters were set and categorized:

Distances

dI_i - Distance from zone i to both the starting and ending point of the route associated with the zone, in meters.

dB - Distance associated with the pathing along the common branch route for the side zones, in meters.

Auxiliary

$P_i = \sum_{j=0}^J \left(\frac{X_{ij} * S_j * QZ_i}{Z_i * QP} \right)$, $\forall_i$ - Demand of product, in product units, with the weight factor taken into account, allocated to each zone already considering the demand balancing factor explained

earlier. This parameter is auxiliary to help the understanding of the constraints, as it could be substituted by the corresponding expression when it appears.

$PT_i = \sum_{i_2=0}^i \sum_{c=0}^C (Y_{ic} * Y_{i_2c} * P_{i_2})$, $\forall_i$ - Total demand, in product units, with the weight factor taken into account, of every zone that belongs to the same cluster as zone i , up to zone i . Auxiliary parameter to help with the formulation.

$NRT_i = \sum_{i_2=0}^{i-1} \sum_{c=0}^C (Y_{ic} * Y_{i_2c} * NR_{i_2})$, $\forall_i$ - Sum of routes that return in every zone that belongs to the same cluster as zone i , up to zone $i - 1$. Auxiliary parameter to help with the formulation.

The term $\sum_{c=0}^C (Y_{ic} * Y_{i_2c})$ assures that only the parameters and decision variables associated with the same cluster as the one of the current zone are considered. It acts as a dividing factor to allow *return* routes to be calculated for different clusters in the same model.

Objective function

Minimize the total distance travelled:

$$\sum_{i=0}^I (NR_i * dI_i * 2) + \sum_{i=0}^I (NS_i * dS_i) + NB * dB - (1b)$$

The first term in expression (1b) replaces the total picking distance, which for this model utilizes the number of routes that return in each zone and the distance associated with the picking of said route. The second term, linked with the total shipping distance, remains the same, but a third term is added, which relates to the total distance travelled in the common branch associated with the picking of the side zones, which is specific to deal with the case study layout.

Constraints

The following new constraints were set:

$$NR_i \geq \frac{PT_i}{QP} - NRT_i, \forall_i - (4b)$$

$$NB \geq \sum_{c=2}^C \sum_{i=0}^I (Y_{ic} * \sum_{j=0}^J (\frac{X_{ij} * S_j * QZ_i}{QP * QP * Z_j})) - (9)$$

Constraint (4b) replaces constraint (4a) from the first model, so that the new routing is calculated for the picking distances. The expression $\frac{PT_i}{QP}$ in the constraint gives the total amount of routes needed to supply every zone up to the current one, i , with the division by clusters and the demand balancing factor already taken into account. By subtracting from the total amount of routes needed the amount of routes that already returned up to the current zone the number of new routes that return in this zone is achieved. This is repeated for every zone until every demand is satisfied.

Constraint (9) determines the amount of total times the routes are performed to the side zones, using a method similar to the one for determining routes to clusters in the first model, but only considering the zones that belong to clusters in the left side.

4.4 Model Extension (More Accurate Demand)

To avoid the demand balancing simplification done in the models above, an extension that can describe the demand of each zone more accurately and that works for both models was proposed.

Since it was a complex task to figure out an even division of demand through multiple zones belonging to the same store, due to the different capacities of the zones and the need to keep the models linearity, the proposed solution breaks down every zone into sub-zones, or pallets, all of the same capacity. The created sub-zones are processed by the model in the same way as zones, with the same constraints and parameters, although the new index for zones obviously enlarges. This way a even distribution through the used pallets can be reached, based only on distance.

Every sub-zone needs to have the same parameters for distance and belong to the same cluster as the original zone, and since sub-zones are not unique, meaning some of them actually belong to the same original zone, an additional constraint needs to be set, that guarantees only the same store can be allocated to all of the related sub-zones. Since the simplification in the model above utilized the zones capacity, and every new sub-zone has the same capacity, the simplification does not need to be changed or removed, and ends up behaving as the identity property of multiplication, multiplying by one.

4.4.1 Mathematical Formulation

To translate the details of sub-zones that belong to the same zone, the following parameter and constraint need to be added.

Parameters

$B_{ii2} \in \{0, 1\}$ - 1 if sub-zone i belongs to the same zone as sub-zone i_2 .

Constraints

$$X_{ij} \leq \sum_{i2=0}^I (B_{ii2} * X_{i2j}), \forall_i : \sum_{i2=0}^I (B_{ii2}) \neq 0, \forall_j - (10)$$

For every zone, a chosen sub-zone acts as a master to all others, that are characterized by only being able to keep products for the same store.

Constraint (10) assures that a slave sub-zone can only have the same store allocation as the corresponding chosen master sub-zone.

The parameter above was implemented in a different way in the *Python* script, with an array that had $B_i = i2$ for sub-zones that were slaves and $B_i = -1$ for master sub-zones. It is important to correctly implement that the master sub-zones are not a part of this constraint, or the model would not be feasible, since they don't depend on any other sub-zone's store allocation.

The sequencing of sub-zones is important due to the successive zone placement for the same store, and after a solution is reached, some data conversion is needed, as shipping pallets and routes allocated to the sub-zones need to be attributed to the corresponding zones.

Chapter 5

Instance designing and processing

An instance is considered to be a set of parameters that can be given to a model, representing a specific set of conditions the warehouse is operating under on each day. Creating adequate instances, inspired by real life case study data, is vital to achieve meaningful results from solving the proposed models. Some of the parameters are translated from data acquired in the case study, while others more speculative about how the warehouse operates, but could be easily changed into better suited alternatives if differences were measured or proven.

During the development of the models and during a first phase of result analysis, smaller instances were used to check the accuracy of results and if behaviour was as intended. These instances bypass some of the information gathering steps mentioned below, since they are made from scratch instead of originating from data.

5.1 Demand Assessments

There is known data for the orders of each specific store to the distribution center, for a week of time. This data comes in an *Excel* sheet, with each line representing a part of the order, in a format such as "order date; store number; item id; number of items; item class". Some of these terms are self explanatory, such as "order date", or "store number", representing the store making the order, but its worth explaining that "item id", a number representing each specific product is not used in the model, as "item class" takes priority, since it translates the different types of products in a better way.

From this data, some parameters and indexes can be retrieved using a *Python* script. The number of different stores, throughout each day, as well as the number of different product types are counted. The demand for a day is also retrieved, bundling up items of the same class and store to reach the format chosen for the model input to this parameter, which is store number followed by the demand for each class of product for that store. These parameters can be retrieved for different days, resulting in different instances to be solved, compared and analysed.

5.2 Distance Measurements

Data on the layout to be considered also exists, with coordinates for each zones position and also reception and expedition areas being shown in different lines in an *Excel* sheet. The coordinates can be mapped out to give a more accurate representation of the layout than the one shown in the above chapters, as seen in figure 5.1.



Figure 5.1: Layout using the given zone coordinates

From this data it is possible to count the number of total zones and determine the shipping distances, by calculating the distance from each zone to the expedition point. Reaching the distances associated with the picking routes, sequencing the zones and respective distribution in clusters, is a bit more complex, as this depends on which model is used and on the number of clusters to be used. The information given about the layout also details the size of the zones, which differs based on the section of the zone, which can also be seen in the figure.

For the first model, based on the "U-shaped" routes, there are two clusters with the center and right side zones of the warehouse, being split by the existing gap. Calculating the associated picking route distance is done manually by considering how many zones are in each cluster, the size of the zones and the size of the gap. For the left sided zones different amount of clusters will be considered and compared in the result analysis chapter. Again, the distance of the routes associated with each cluster are calculated manually considering the amount of zones belonging to the cluster and the point where the route integrates with the main "U-shape" to return to the reception area, as explained in the case study routing. The zones are sorted based on cluster, to simplify the creation of the parameter which represents which cluster they belong to.

As for the second model, based on the *return* method, some different steps need to be taken. The shipping distance remains the same, and the number of clusters is fixed for this method. Clusters are split differently though, with the center zones being divided in the middle vertically, into left and right. The right side zones resulting from the split belong to the same cluster as the right side zones of the warehouse. Each zone in these clusters has then a distance to the reception area. The left side zones of the warehouse are split based on which corridor can be used to access them. The distance for picking in these zones is based on the distance from the zone to the main corridor, with another parameter being set for the distance to use in the main corridor.

The sequencing of the zones is vital to this model, so it is crucial to rank them from longest in picking distance to shortest, inside each cluster.

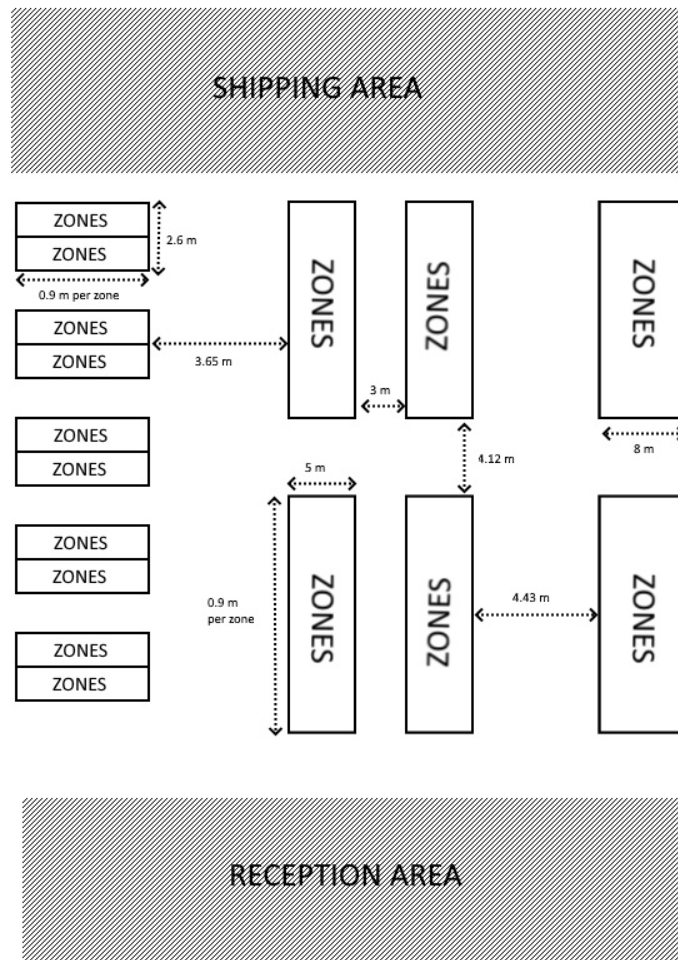


Figure 5.2: Layout with distance measuring

5.3 Quantity Related Assessments

Detailing the quantities and relating factors was much more speculative, since no data was given, as such data is hard to create or measure. Even considering the existence of different product types, calculating how many products of each type can fit into a single pallet is difficult, as in reality the stacking in the pallet is not straight for every product. To deal with this, a weight was created for each product type, to differentiate their relation with the maximum quantity of product in a pallet in a linear way. For early tests it was assumed the weight was the same for every product class, as this simplifies the instance and the model, but in the final tests the following relation was set:

- Weight of Class 1 = 2 * Weight of Class 2

- Weight of Class 2 = 3 * Weight of Class 3

These weights are simple to thinker with though, so no definite relation was chosen. The maximum quantity of products in each pallet, considering the weights, was set to be 30 product units, but again this value is not fixed, although it is hopefully realistic. It was stated that different pallets would be used for picking and shipping, and despite the fact that the model is built to consider this difference, the instances were started with having the same quantity for shipping and picking pallets.

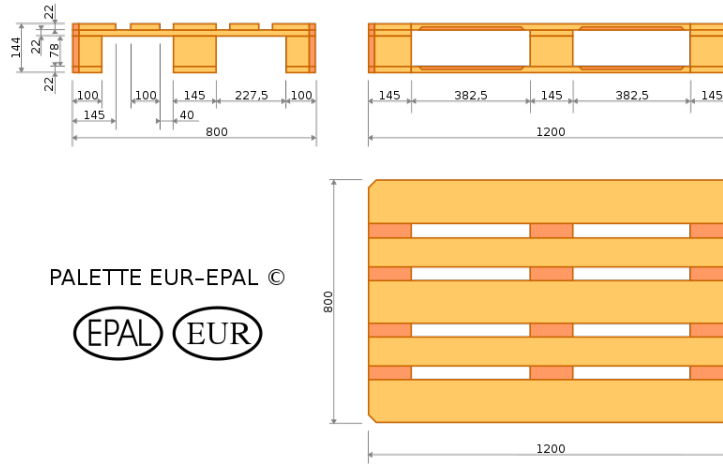


Figure 5.3: Standard European Pallet Sizes [2]

The standard European pallets are used in the case study, which measure to 80x120 cm, and zones have a 10cm border to them. Considering this, and the data that exists of the layout, the amount of pallets that can fit into each zone and the resulting capacity of the zone were determined, with:

- Capacity of the left side zones being one pallet = 30
- Capacity of the middle zones being four pallets = 120
- Capacity of the right side zones being six pallets = 180

A final parameter is necessary, to represent the number of pallets that are needed for each store, to allow the balance of demand between them and keep the model linear. This is done by rounding up the result from the division of the capacity necessary for each stores order and the pallet capacity. This parameter is then counter balanced for zones with bigger capacity, as is considered in the model.

5.4 Instance processing

An *Excel* sheet was prepared with all the data needed and extracted from the other sheets through *Python* scripts. The process of preparing all the indexes and parameters, necessary for the model, to be loaded is mostly manual, since different models and instances were involved with different characteristics and sizes. This could be an area to improve if needed, as the rough

process, which helped in a debugging phase of the program and in keeping the solving time limited when it came to getting results to analyse. For this purpose, a computer with a 6 cores *AMD Ryzen 5 3600* and 16 gigabytes of *RAM* was utilized. When a solution is reached the solver presents information on the decision variables, objective value and can also allow the determination of the optimality gap, based on a best bounds solution it provides. After running the scripts, the resulting decision variables and objective value are printed into the command line and an output *Excel* sheet, from which the results are extrapolated and shown in correlation with the layout.

Each zone is represented by two cells as can be seen in 5.6. The right cell for horizontal zones and the bottom cell for vertical ones represent which store is allocated to the zone, and white -1 means a blank zone, with no associated store. The left/top cell has the amount of shipping pallets needed for the respective operations regarding that zone. Picking route information could also be shown from the model's solution, with how many routes of each type took place for the first model and, for the second model, a number that represents how far into an aisle/corridor operators returned on a route. Additional information could be presented, such as the objective value, total distance travelled in meters, the optimality gap and the split between shipping and picking operations.

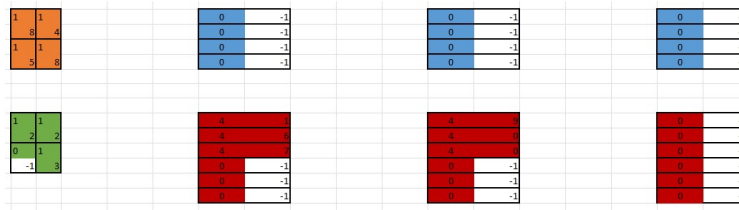


Figure 5.6: Example result representation

Lastly, when visualizing the results, stores are numbered sequentially from zero to the highest number possible for a store. This is a direct consequence from running the data through the mathematical formulation, which needs stores to be numbered this way. The first conversion is done when translating the data relating to the demand, with the store number being sequenced by order of appearance in the *Excel* sheet lines. The inverse conversion is possible through a script, but since the actual store is not of interest to these results, it was not done. If these models were to be applied to the real world, or integrated with other simulations that did not use the same ordering, this conversion would become necessary.

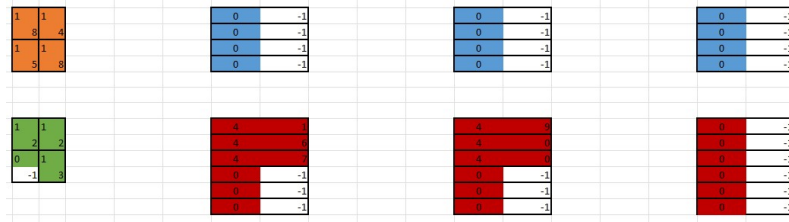


Figure 6.2: Model 1 result representation without extension

In figure 6.2, it can be seen that the model favors the routes that utilize the layout gap as a passage to cross the aisles to routes that utilize the full length of the warehouse. Inside the chosen clusters, the shipping distance is what dictates the best placement, with stores being allocated to zones closer to the shipping point. Since, without the extension, the placement of stores throughout the zones takes into account zone capacity, zones whose capacity were closer to the space needed for a store's order demand were preferred, even if they were tied to slightly longer routes, as can be seen by the usage of the left zones.

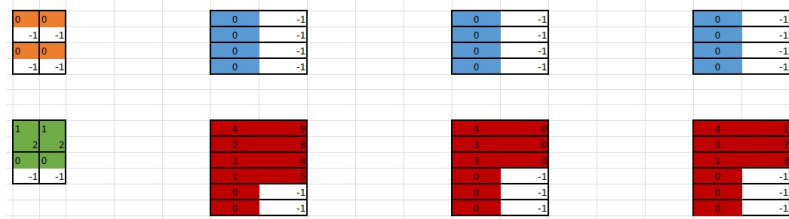


Figure 6.3: Model 1 result representation with extension

While looking at the first model 6.3, but with the extension that describes demand more accurately, it can be seen that the placement still favors shorter routes, with shipping distance still coming into consideration. The difference that can be spotted from the first result, is that the model places some of the stores with lower demand on zones with exceeding capacity, since the distance is the only concern now, resulting in the moving of some stores placed on the left side to the middle and right aisles. The usage of shipping pallets also shows the difference in demand calculation for each zone.

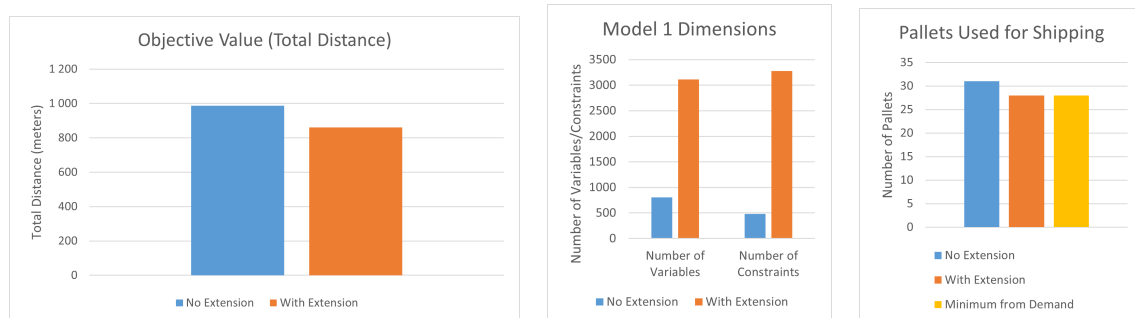


Figure 6.4: Extension comparisons for Model 1

When comparing both experiments, arguments can be made for and against the use of the extension. With the extension, the objective value drops and the order demand is more accurately represented, which is measured by using the number of pallets for shipping and comparing with the total number of pallets needed for all stores in the parameters. On the other side, without the extension, it can be seen that the number of variables and constraints is kept lower, which keeps the model simpler and shrinks the optimal gap reached in the same amount of time from around 3% to 0%.

6.1.2 Return routing

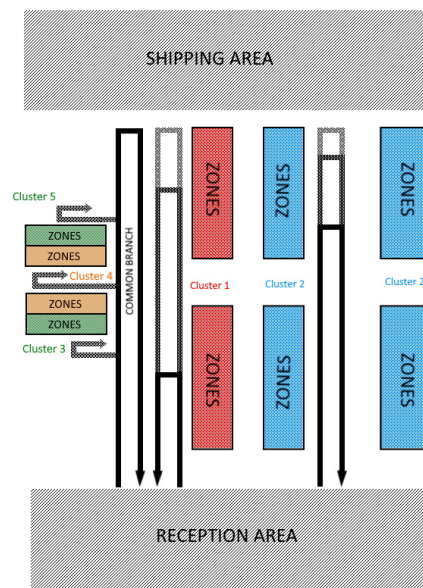


Figure 6.5: Model 2 picking route map

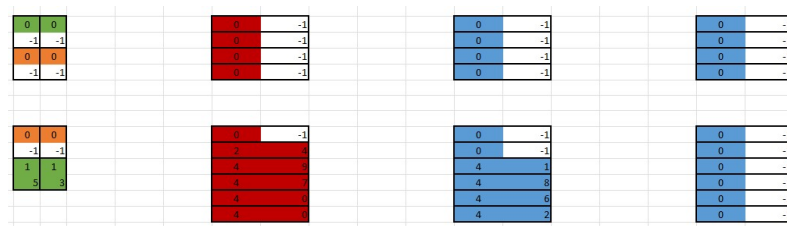


Figure 6.6: Model 2 result representation without extension

From 6.6 the differences between models are easily perceptible, as stores are placed closer to the reception area, to avoid extending the length of routes unnecessarily. Since the picking routing considers that the operator must travel to the zone and back, but for the shipping distance only the travelling from the zone to the destination is taken into account, stores are placed in a way

to minimize picking distance the most. In the first model picking also had more weight in the decision, but due to the fixed picking routes this is less discernible. The effect from the demand balancing factor is again noticeable, as some stores are placed in the left side and no stores are placed in the right side, on account of zone capacity. It is interesting to verify the equilibrium between matching zone capacity and picking route length in the allocation process.

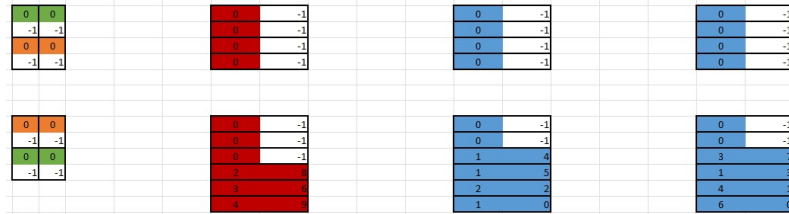


Figure 6.7: Model 2 result representation with extension

In 6.7, utilizing the extension, the differences are noticeable, as again the model starts deciding only based on distance and not on zone capacity. The right side is now utilized, as routes can simultaneously supply two sides of the aisle, economizing on picking pallets. The behaviour of placing stores closest to the reception area stays the same. Due to the impact that placement can have on the picking routes over-weighing the importance of the shipping distance, no zones are attributed to the left side.

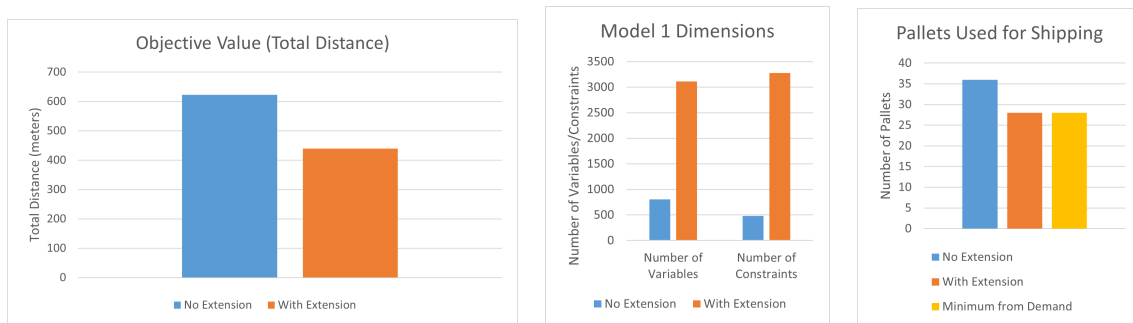


Figure 6.8: Extension comparisons for Model 2

Again, the differences between using or not using the extension can be seen when comparing both results. Overall, including the results from both models, there is a relation that can be spotted from the rise in estimated demand and the objective value, in percentage. For both models, when using the extension, the demand attributed to zones that can be spotted by counting shipping pallet usage is equal to the estimated actual demand from the parameters and data inputted. The same cannot be said for the simplification that comes from not including the extension, as the first model sees a 11% overestimation in demand, which results in a 15% rise in objective value. As for the results from the second model, the demand overestimation with no extension is of 29%, which results in a 42% jump in objective value. This big difference in overestimation can be justified by

understanding that model 2 favours zones with higher capacity than model 1, which evolves into a bigger error in the balancing factor.

The trade off is that using the more accurate extension demands the usage of a higher number of variables, a growth of around the 288% mark when compared to the simplification. The same happens for constraints, with a rise of near the 570%. These added variables and constraints strain the model, meaning that the solver takes longer to reach solutions. When using these smaller test instances the difference between running the software solver for both could already be noticed, with the use of the extension taking longer to run, needing an added time limit, of about five minutes, since the optimal solution could not be reached hastily, even though the same model without extension could reach it quickly. So, even if the extension guarantees better results, it presents a higher optimality gap, in the 3% range, and takes longer in the solving process.

6.2 Case Study Instances

The following results were achieved using instances based on the case study data, which were assembled based on the methods described in section 5.4. The case study has 561 zones and above 100 stores to place, 154 specifically for the first day of data, which the instances were based on.

6.2.1 Assumptions and Simplifications

With the scaling in dimension from the base instances to the case study based ones, questions about the time it takes the software to reach a solution arise.

Since it would take hours for the software to understand the variants and constraints in order to reach a satisfying solution with instances of this order, with the current computing power, some of the non essential conditions were relaxed. As was discussed above in section 6.1, the extension that describes demand more accurately brings a growth in complexity, so it was not used for these instances. Although it can be hard to predict how the extension would impact the solutions, it is expected that due to the many orders it would be easier to fit stores in adequate sized zones, lowering the imprecision caused by the simplification in demand estimation, that can still be measured by comparing the number of shipping pallets used to the data from the store orders. Constraints (5), (6) and (7), which relate to keeping zones belonging to the same store in successive zones, were also removed, as this shortens the solve time substantially, and its impact on the comparison between solutions is minimal.

As can be seen in figure 6.9, comparing the relaxed version of the first model to the one with the use of the extension and constraints (5), (6) and (7), a growth of 427% is spotted in the number of variables, and the number of constraints with the extension usage is 360 times higher than the relaxed counterpart, which results in hugely longer solving times. The estimation of demand only results in a 2.5% rise over the data from the orders. These results were taken from an instance based on the first day of orders, and only using one route to supply the left sided zones.

The relaxed model was given 20 minutes to reach a solution, time length which was kept for the other instances. Most of the results reached were not optimal solutions, so a non-zero optimality

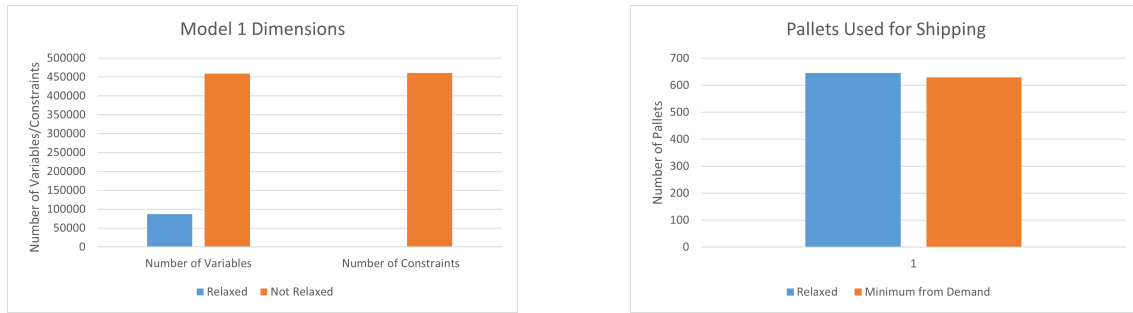


Figure 6.9: Relaxed version comparisons for Model 1

gap exists. This correlates to some stores having a clear placement improvement available to the eye, but which would take the model more time to reach. Further optimization stages, with resource to heuristics or searching neighbour solutions to the one reached could be created to improve this.

6.2.2 Shipping Effect

To study the effects the shipping operations add to the allocation of stores, the same instance was compared with and without them. The instance used was based on the first day of orders, and utilizes only one route to supply the zones on the left side of the warehouse, as can be seen in the map of the picking routes [6.10](#).

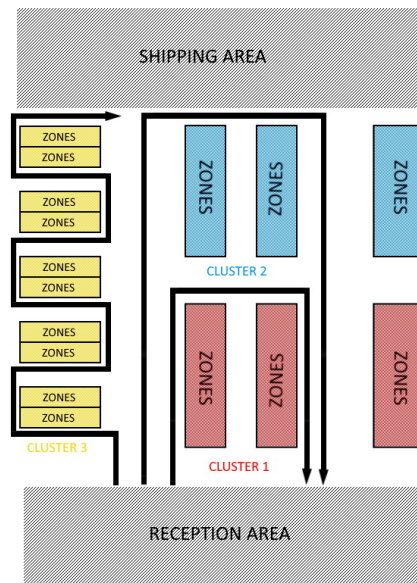


Figure 6.10: Model 1 picking route map

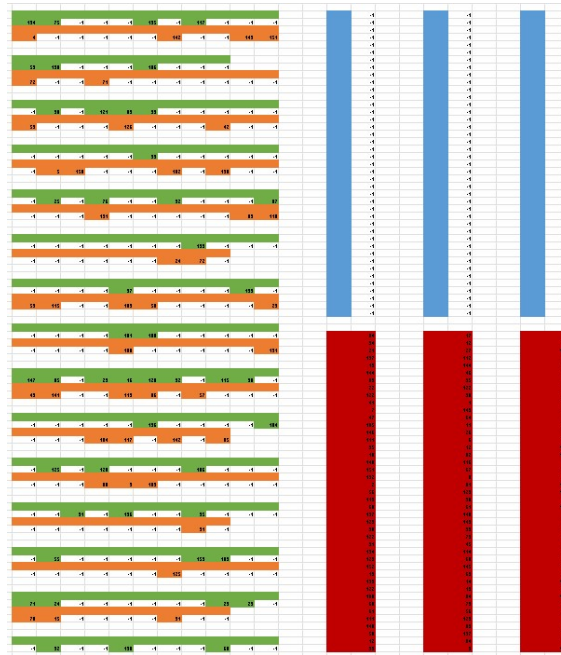


Figure 6.11: Result representation without shipping

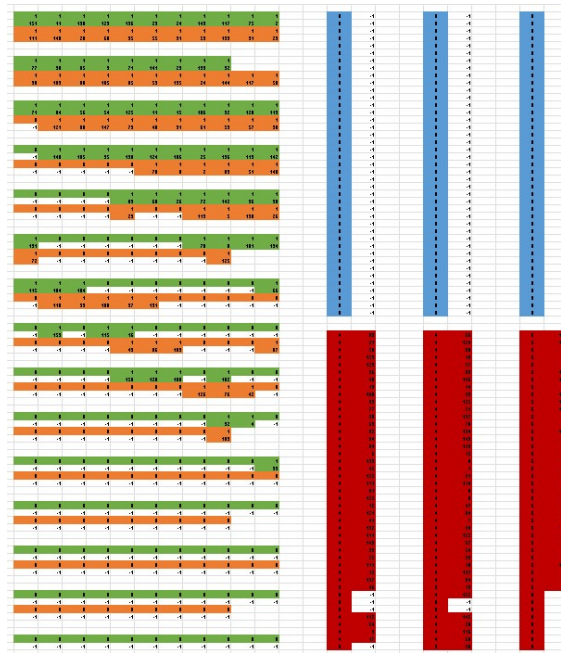


Figure 6.12: Result representation with shipping

Without shipping operations [6.11](#), the model fills the cluster associated with the shortest picking distance completely, and has excess stores placed on the left cluster, since it minimizes the demand overestimation, even if it has a slightly longer associated picking route. On the left cluster, stores are spread almost randomly, since the picking distance is considered the same for all

due to the routing and no shipping distance is present.

On 6.12 the impact of the shipping distance can be noticed, as stores are disposed on the left cluster closer to the shipping area. Some zones also moved cluster, since the shorter shipping distance and demand overestimation compensates the longer picking route. A 3% optimality gap is left, which can be clearly seen, since some stores could be moved to zones closer to the shipping point, inside the same cluster.



Figure 6.13: Total distance comparison with and without shipping

Adding the shipping operations to the model increases the distance travelled in the picking operations by 3.6%, 6.13. For this instance, the shipping operations would represent 33% of the total distance travelled, with the picking operations representing the rest.

6.2.3 Cluster Analysis

This subsections aims to understand the impact of dividing the left side of zones into different clusters. On the real warehouse, the routing can be more dynamic, with operators choosing to continue the *S-shaped* routing or return to the reception area at each point, but for the purpose of model 1, a strict division of the zones and routing was done, as was explained in the model concepts. Instances that represent having just one cluster, two clusters split in order to have one route pass to the gap on the middle aisles, and fifteen clusters, keeping each group of zones in a unique cluster, were made.

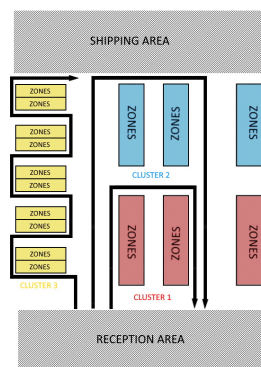


Figure 6.14: Model 1 picking route map using one cluster

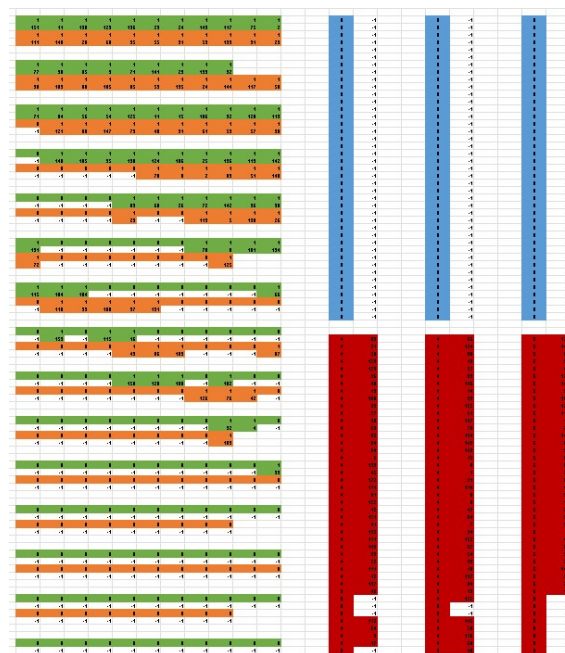


Figure 6.15: Result representation using one cluster

As can be seen in figure 6.15, if the left group of zones is kept a single cluster, whose routing finishes through the top of the warehouse, the stores are placed closest to the shipping area, with a few exceptions that can be attributed to the optimality gap.

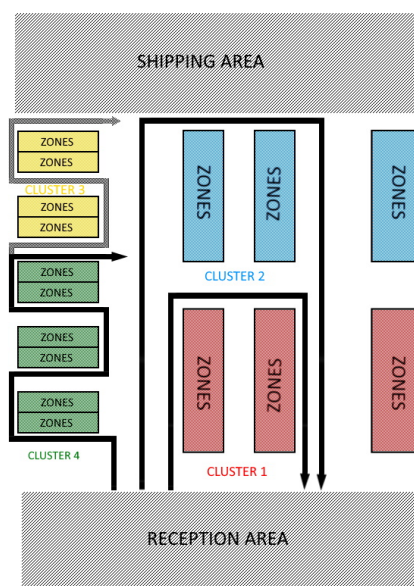


Figure 6.16: Model 1 picking route map using two clusters

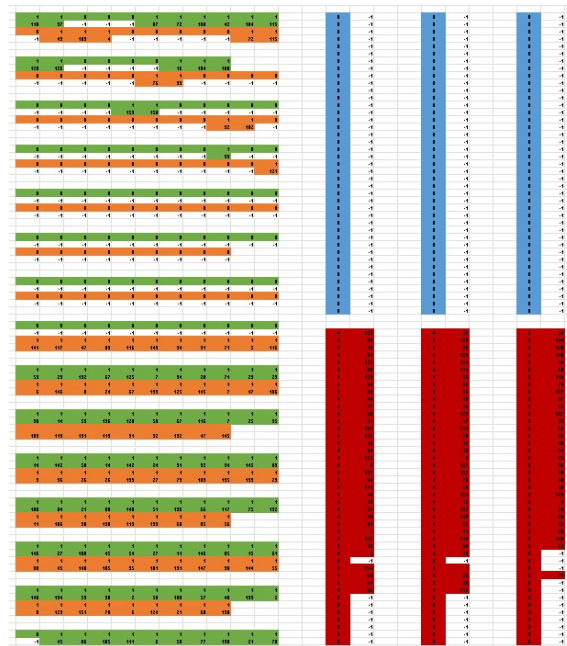


Figure 6.17: Result representation using two clusters

If the left sided zones are split into two clusters, 6.17, to allow one picking route to cross through the middle gap existing in the center aisles, it is noticeable that most stores are placed in favour of this new picking route. Some of the stores that were allocated to center zones beforehand even swap to the new cluster, since the size of their order is closer to the capacity of these zones. Some stores still exist in the top of the warehouse, seeing as the zones associated with the shorter picking route were already full.

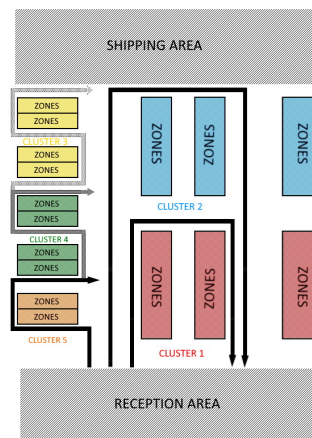


Figure 6.18: Model 1 picking route map using fifteen clusters

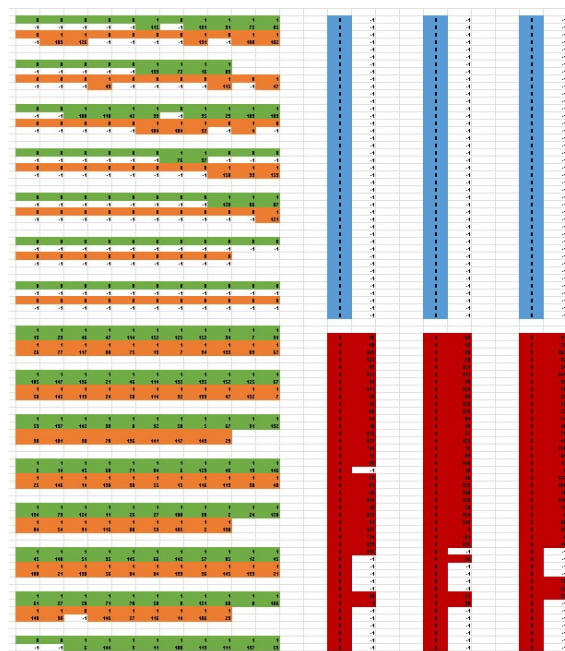


Figure 6.19: Result representation using fifteen clusters

Analysing the placement when using fifteen clusters, 6.19, shows similar results to the usage of two clusters. In the left side, stores are allocated to clusters whose picking routes can cross through the middle gap first, and then excess stores are placed on the top of the warehouse, closer to the shipping point. One specific row in the middle of the warehouse notices changes, since the picking routing associated with it uses the middle gap when the zones are divided into fifteen clusters, but has to utilize the top of the warehouse when the two cluster split happens.

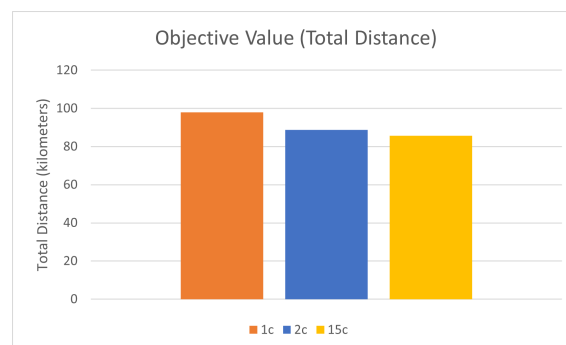


Figure 6.20: Total distance comparison with different cluster divisions

When comparing the total distance travelled using the different mapping of clusters, 6.20, it is shown that the instance with most clusters minimizes the goal the most. Utilizing one cluster does not represent the reality of the operations to a full extent, since having the shortcut in picking routes is a key detail of this case study. Between choosing the division into two or fifteen clusters, more analysis is necessary. The number of variables and constraints between them is similar, but

the optimality gap rises from 3% to 5% when increasing the number of clusters. This results in a 3% decrease in the objective value, but since the actual store allocation remains alike, and the decrease happens because of the changes in the modeling of the picking routing, it is not certain that using more than two clusters is worth the added percentage in optimality gap.

6.2.4 Aisle Gap Changes

When it comes to enhancing the current operations while keeping the same routing, a possible change is to move the gap that splits aisles up or down.

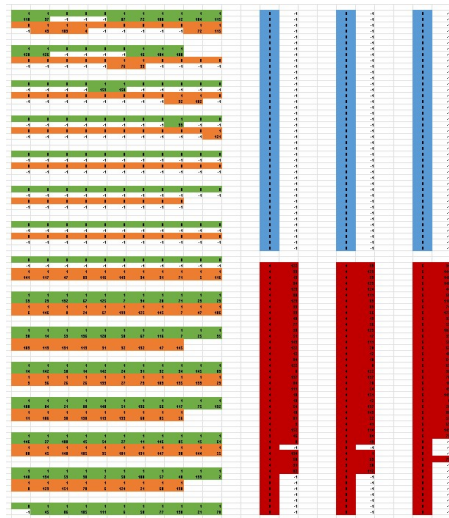


Figure 6.21: Result representation using the current gap

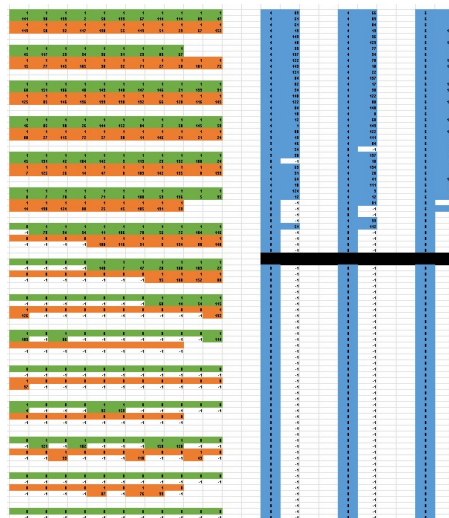


Figure 6.22: Result representation using no gap

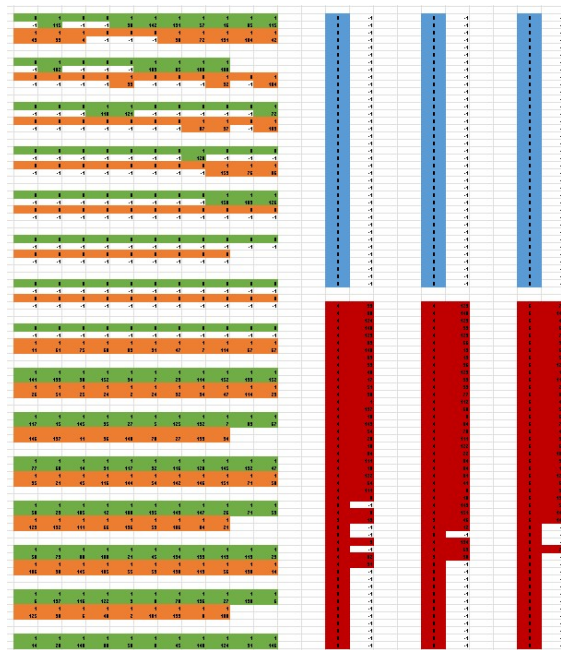


Figure 6.23: Result representation moving the gap up

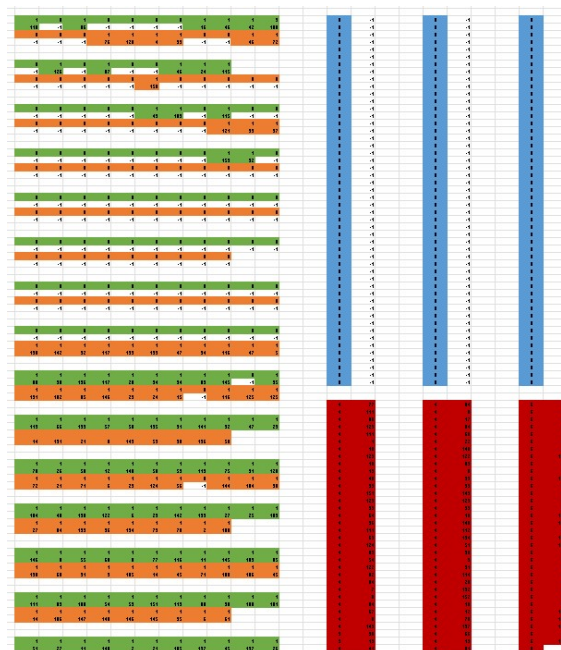


Figure 6.24: Result representation moving the gap down

As can be seen in the figures above, not utilizing a gap on the layout only increases the total distance travelled, as the picking route shortcut stops existing, making the routes even more limited. For the store order data of this specific day, it can be seen that moving the gap down, meaning less stores are in the bottom cluster but the associated route is also shorter, proves advantageous,

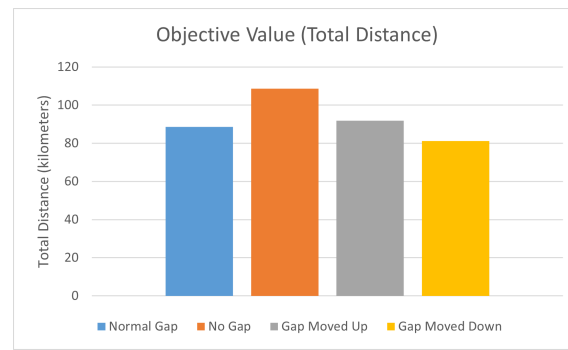


Figure 6.25: Total distance comparison with different layout gaps

by almost a 9% total distance save, as the store to cluster allocation remains the same and the picking route shortens. Moving up the gap creates more empty space and makes the picking route longer.

A deeper study would have to be made, utilizing orders regarding different days, in order to reach solid conclusions about where the gap should actually be placed. Due to the highly uncertain demand environment though, it is hard to reach average store to cluster allocation and find the best location. The use of the extension could also impact the cluster allocation and change the results obtained, so this tool is not ideal at the current state to reach conclusions about the gap.

6.2.5 Routing Analysis

The two different models created represent two different types of routing that could be possible in the distribution center. The results shown in the above sections all utilize the current routing, *S-shape*, which is easier to use by operators and does not have congestion problems.

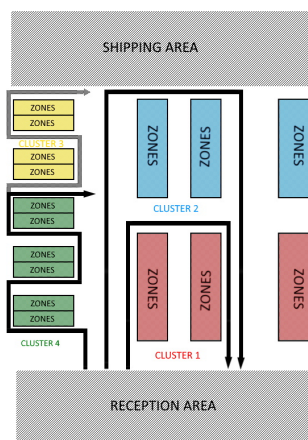


Figure 6.26: Model 1 picking route map

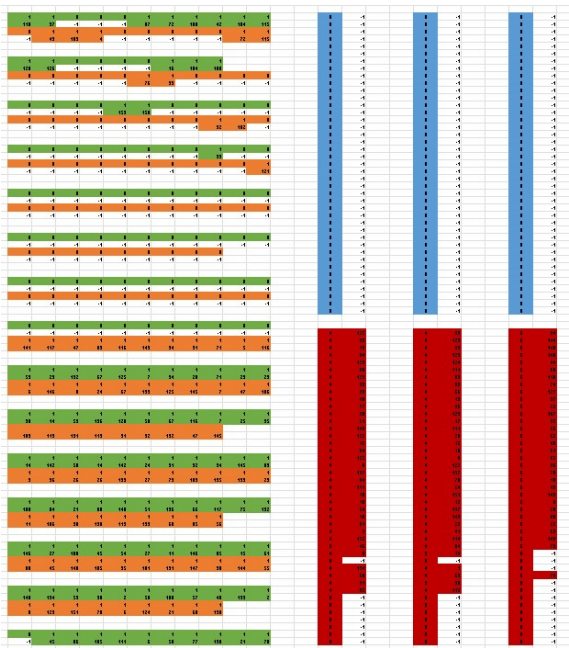


Figure 6.27: Model 1 result representation

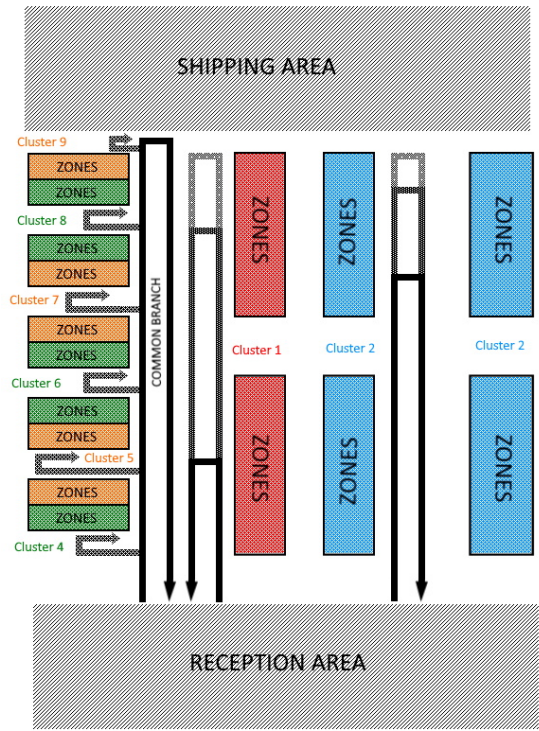


Figure 6.28: Model 2 picking route map

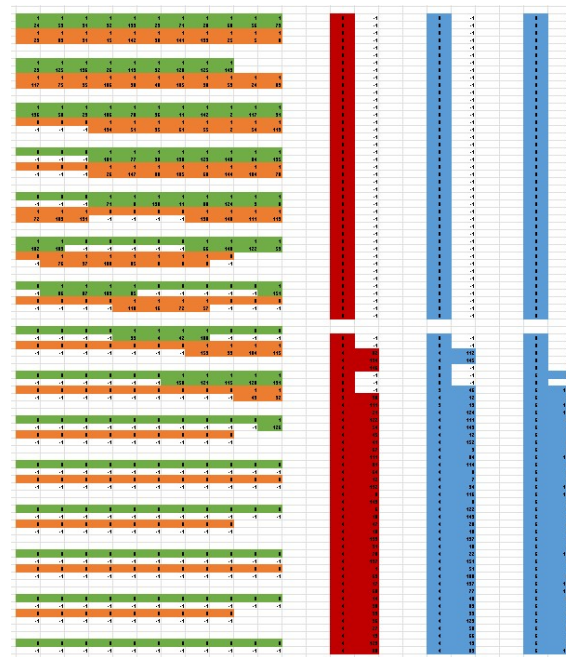


Figure 6.29: Model 2 result representation

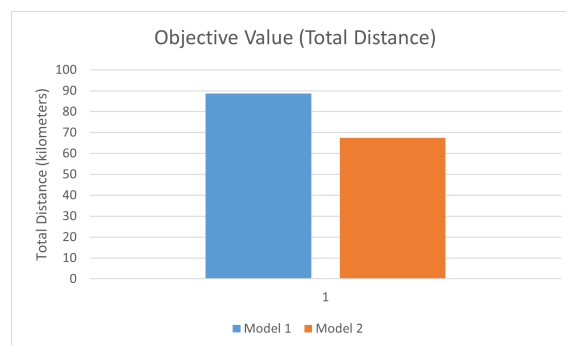


Figure 6.30: Total distance comparison with different routing methods

Looking at 6.29, its perceivable that in the center aisles the placement of stores is made closer to the reception area, in order to shorten the picking routes, that take priority over the shipping distance. Due to the fact that the extra step of calculating another *return* method for the different groups of left zones was not taken, the placement of stores is made according to shipping distance again.

The *return* method shows a 25% decrease in total distance over the *S-shape*, but this result can not be taken as certain. Since the models represent different routing types it is hard to have a direct comparison, especially because the congestion caused is not taken into account. In optimal conditions the *return* routing is timed such as no congestion is caused, but this might not be possible, causing delays that do not increase the distance travelled but affect the operations. Integration

of the results reached with a simulation environment could help understand how the congestion affects the warehouse operations.

Chapter 7

Conclusion

The main goal of this dissertation was to design and test a model to optimize day-by-day operations at a distribution center, by configuring the allocation of stores to picking layouts, considering product demand and internal logistics. A tool was developed that allows to, based on instances, reach a satisfying arrangement of zones to stores.

The objective of minimizing the total distance travelled in the center during the operations was established, and although it does not incorporate every important factor to the operations, it gives a measurable total to be able to compare methods and solutions, and ultimately enhance the efficiency of the operations. The other key factors are related to timing issues, such as truck arrival timing, number of workers and congestion, which were harder to consider under the mathematical programming lenses.

With the aim of minimizing the set objective, two different models were created, based on two methods of routing, *S-shape* and *return*. Some changes inside the routing, such as the division of zones into different clusters, and small differences in an existing gap in the layout, were studied to see their effect on the result. The same was done to the shipping operations, to understand how they affect the total distance travelled and the store placement inside the warehouse.

Some of the underlying conditions of the distribution center were simplified in the modeling, and this could be studied further, such as the shipping operations and the truck load optimization. A different method of preparation for the picking pallets could also have impact on the routing. The estimation of demand for every zone was challenging, since some stores could expand to more than one zone, which lead to the creation of an extension which more accurately describes the demand split.

The developed tool relies on *Excel* sheets as inputs of the parameters and the sizes of sets, and those sheets were prepared manually. A more dynamic automatic tool that could calculate picking route distances and shipping distances based on a given layout of the warehouse could be an improvement, even if the specific layout and instances were built for a unique case study. Integration with a simulation based approach could be the next step for this project, since it would allow the contemplation of the impact of new conditions, such as congestion and other timing delays.

To conclude, this dissertation allowed the development of a tool that attributes stores to zones in a distribution center, and permitted a study on how different changes would impact the overall objective. The results analysed were based on a real case study, but with some adaptation the model could fit other centers. Some further development can be made, to improve the tool's interface and allow the integration with a simulation to consider other factors.

References

- [1] René de Koster, Tho Le-Duc, and Kees Jan Roodbergen. Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, 182(2):481–501, 2007. doi:[10.1016/j.ejor.2006.07.009](https://doi.org/10.1016/j.ejor.2006.07.009).
- [2] Wikipedia contributors. Eur-pallet — Wikipedia, the free encyclopedia, 2023. [Online; accessed 6-June-2023]. URL: <https://en.wikipedia.org/w/index.php?title=EUR-pallet&oldid=1148482174>.
- [3] Mário Amorim-Lopes, Luís Guimarães, João Alves, and Bernardo Almada-Lobo. Improving picking performance at a large retailer warehouse by combining probabilistic simulation, optimization, and discrete-event simulation. *International Transactions in Operational Research*, 28(2):687–715, 2020. doi:[10.1111/itor.12852](https://doi.org/10.1111/itor.12852).
- [4] Ammar Y. Alqahtani. Improving order-picking response time at retail warehouse: a case of sugar company. *SN Applied Sciences*, 5(1), 2022. doi:[10.1007/s42452-022-05230-6](https://doi.org/10.1007/s42452-022-05230-6).
- [5] Fabrizio Dallari, Gino Marchet, and Marco Melacini. Design of order picking system. *The International Journal of Advanced Manufacturing Technology*, 42(1-2):1–12, 2008. doi:[10.1007/s00170-008-1571-9](https://doi.org/10.1007/s00170-008-1571-9).
- [6] Johan Oxenstierna, Jacek Malec, and Volker Krueger. Efficient order batching optimization using seed heuristics and the metropolis algorithm. *SN Computer Science*, 4(2), 2022. doi:[10.1007/s42979-022-01496-0](https://doi.org/10.1007/s42979-022-01496-0).
- [7] Sebastian Henn. Algorithms for on-line order batching in an order picking warehouse. *Computers Operations Research*, 39(11):2549–2563, 2012. doi:[10.1016/j.cor.2011.12.019](https://doi.org/10.1016/j.cor.2011.12.019).
- [8] Elizabeth Jewkes, Chulung Lee, and Ray Vickson. Product location, allocation and server home base location for an order picking line with multiple servers. *Computers Operations Research*, 31(4):623–636, 2004. doi:[10.1016/s0305-0548\(03\)00035-2](https://doi.org/10.1016/s0305-0548(03)00035-2).
- [9] Charles G. Petersen. An evaluation of order picking routeing policies. *International Journal of Operations Production Management*, 17, 1997. doi:<https://doi.org/10.1108/01443579710177860>.
- [10] Yixuan Su, Xi Zhu, Jinlong Yuan, Kok Lay Teo, Meixia Li, and Chunfa Li. An extensible multi-block layout warehouse routing optimization model. *European Journal of Operational Research*, 305(1):222–239, 2023. doi:[10.1016/j.ejor.2022.05.045](https://doi.org/10.1016/j.ejor.2022.05.045).

- [11] Charles G. Petersen and Gerald Aase. A comparison of picking, storage, and routing policies in manual order picking. *International Journal of Production Economics*, 92(1):11–19, 2004. doi:[10.1016/j.ijpe.2003.09.006](https://doi.org/10.1016/j.ijpe.2003.09.006).
- [12] Teun van Gils, Katrien Ramaekers, An Caris, and René B. M. de Koster. Designing efficient order picking systems by combining planning problems: State-of-the-art classification and review. *European Journal of Operational Research*, 267(1):1–15, 2018. doi:[10.1016/j.ejor.2017.09.002](https://doi.org/10.1016/j.ejor.2017.09.002).
- [13] Jan-Erik Justkowiak and Erwin Pesch. Stronger mixed-integer programming-formulations for order- and rack-sequencing in robotic mobile fulfillment systems. *European Journal of Operational Research*, 305(3):1063–1078, 2023. doi:[10.1016/j.ejor.2022.06.051](https://doi.org/10.1016/j.ejor.2022.06.051).
- [14] Zhengcai Cao, Lijie Zhou, Chengran Lin, and Mengchu Zhou. Solving an order batching, picker assignment, batch sequencing and picker routing problem via information integration. *Journal of Industrial Information Integration*, 31, 2023. doi:[10.1016/j.jii.2022.100414](https://doi.org/10.1016/j.jii.2022.100414).
- [15] Eric H. Grosse, Christoph H. Glock, and W. Patrick Neumann. Human factors in order picking: a content analysis of the literature. *International Journal of Production Research*, 55(5):1260–1276, 2016. doi:[10.1080/00207543.2016.1186296](https://doi.org/10.1080/00207543.2016.1186296).
- [16] Robert Nieuwenhuis, Albert Oliveras, and Enric Rodriguez-Carbonell. A heuristic approach to the design of optimal cross-docking boxes. *IEEE Access*, 9:122578–122588, 2021. doi:[10.1109/access.2021.3109976](https://doi.org/10.1109/access.2021.3109976).
- [17] Mahziar Taghizadeh, Amir Abbas Shojaie, Amir Homayoun Sarfaraz, Sadigh Raissi, and Reza Lotfi. A multiobjective mathematical model for truck scheduling problem in multidoor cross-docking system. *Discrete Dynamics in Nature and Society*, 2022:1–15, 2022. doi:[10.1155/2022/2171305](https://doi.org/10.1155/2022/2171305).
- [18] Reza Shahabi-Shahmiri, Sobhan Asian, Reza Tavakkoli-Moghaddam, Seyed Meysam Mousavi, and Mohsen Rajabzadeh. A routing and scheduling problem for cross-docking networks with perishable products, heterogeneous vehicles and split delivery. *Computers Industrial Engineering*, 157, 2021. doi:[10.1016/j.cie.2021.107299](https://doi.org/10.1016/j.cie.2021.107299).
- [19] Jan Van Belle, Paul Valckenaers, and Dirk Cattrysse. Cross-docking: State of the art. *Omega*, 40(6):827–846, 2012. doi:[10.1016/j.omega.2012.01.005](https://doi.org/10.1016/j.omega.2012.01.005).
- [20] Miguel Horta, Fábio Coelho, and Susana Relvas. Layout design modelling for a real world just-in-time warehouse. *Computers Industrial Engineering*, 101:1–9, 2016. doi:[10.1016/j.cie.2016.08.013](https://doi.org/10.1016/j.cie.2016.08.013).
- [21] Tobias Achterberg. Scip: solving constraint integer programs. *Mathematical Programming Computation*, 1(1):1–41, Jul 2009. URL: <https://doi.org/10.1007/s12532-008-0001-1>, doi:[10.1007/s12532-008-0001-1](https://doi.org/10.1007/s12532-008-0001-1).