Sigmund Pedersen

Investigating the Effect of Warehouse Policies on Order Picking Time in Multiple Vertical Lift Modules

Master's thesis in Global Manufacturing Management
Supervisor: Fabio Sgarbossa
June 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Mechanical and Industrial Engineering

NTNU
Norwegian University of Science and Technology

ST. OLAVS HOSPITAL
UNIVERSITETSSYKEHUSET I TRONDHEIM

Sigmund Pedersen

Investigating the Effect of Warehouse Policies on Order Picking Time in Multiple Vertical Lift Modules

Master's thesis in Global Manufacturing Management
Supervisor: Fabio Sgarbossa
June 2023

Norwegian University of Science and Technology
Faculty of Engineering
Department of Mechanical and Industrial Engineering

NTNU
Norwegian University of Science and Technology

Preface

This thesis marks the final culmination of my journey as a student at the Norwegian University of Science and Technology, NTNU. The thesis is written as the final part of the *Global Manufacturing Management* international masters program offered by the Department of Mechanical and Industrial Engineering at NTNU.

I would like to thank my supervisor, Professor Fabio Sgarbossa, for his great guidance and feedback throughout the development of this thesis. He has given valuable insights and advice. Further, I would like to thank Logistikksenter Helse Midt-Norge, LS HMN, for the opportunity to write this thesis in collaboration with them. I would also like to extend a special thanks to Vegard Jensås Andersen, logistics engineer at LS HMN, for providing valuable information from LS HMN, guidance and feedback during the process of writing this thesis.

Furthermore, I would like to express my sincere gratitude to my classmates for their unwavering support, which kept me grounded and prevented any moments of despair. Their camaraderie ensured that the door to Room 225 was always open, ready to welcome me if I ever forgot my keycard. Lastly, I want to express my heartfelt appreciation to my loving girlfriend, whose unwavering presence and unwavering belief in my abilities have been a constant source of motivation and encouragement throughout this journey. Her support and understanding have provided me with the stability and inspiration needed to overcome obstacles and pursue my academic aspirations.

Sigmund Pedersen

June 8th, 2023

Trondheim

Abstract

A warehouse performs various functions and activities, including replenishment, put-away, storage, and order picking, with different policies influencing the execution of these functions and activities. Order picking is the most cost- and time-consuming activity of a warehouse, and up to 55% of warehouse operations costs are related to order picking. Investments in automated solutions and new technology would reduce the order picking time, resulting in lower operations costs. The Vertical Lift Module (VLM) is an automated solution which would lower the order picking time. A VLM consists of an encased pair of racks where trays are stored, with an automatic storage and retrieval (AS/R) device between. The AS/R device extracts trays from the racks and displays them in a picking bay, and after the picking is executed, the tray is inserted back in the racks.

The goals of this thesis are to investigate the impact on order picking time when replenishment and put-away are executed in parallel with order picking, and how changes in the warehouse policies affect the order picking time. Multiple research questions will be answered to achieve these goals: *1)* What are the applicable warehouse policies for a system with multiple VLMs working in sequence? *2)* What is the impact on order picking time in a system with multiple VLMs when replenishment and put-away are executed in parallel with order picking? *3)* What is the impact on the order picking time when changing between the applicable warehouse policies in a system where multiple VLMs operate in sequence? *4)* What is the impact on the order picking time when the number of operators and active VLMs are altered?

The first part of the thesis is a theoretical study of how a warehouse operates, its functions and activities, and different policies regarding these functions and activities. Then follows a literature review to map the current research on VLMs, where there is a clear gap in how replenishment and put-away affect a system with VLMs and how multiple VLMs operating in sequence are subjected to changes in warehouse policies. The third part is a case study of a distribution centre for non-pharmaceutical goods which consists of multiple VLMs where replenishment and put-away are executed in parallel with order picking. Then follows a section where the applicable warehouse policies are discussed. The fourth part describes how a Discrete Event Simulation (DES) model has been used to investigate the effects of altering the warehouse policies.

The main conclusion of this study is that replenishment and put-away do not affect the order picking time or the waiting during order picking. However, the stay time of an order is increased when replenishment and put-away are included. Further, the warehouse policies affect the order picking time, either positively or negatively, depending on the policy. Lastly, when the number of operators increases, the replenishment order execution time increases, while when the number of active VLMs increase, the order picking time is reduced.

Sammendrag

Et lager har flere ulike funksjoner og aktiviteter som inkluderer etterfylling, frasetting, lagring og ordreplukking, med ulike policyer som påvirker utførelsen av disse funksjonene og aktivitetene. Ordreplukking er den mest kostnads- og tidskrevende aktiviteten i et lager, opptil 55% av driftskostnadene er knyttet til ordreplukking. Investeringer i automatiserte løsninger og ny teknologi vil kunne redusere ordreplukkingstiden, noe som resulterer i lavere driftskostnader. Vertical Lift Module (VLM) er en automatisert løsning som vil senke tiden brukt for ordreplukk. En VLM består av to innkapslete reoler med flere hyller med en automatisk lagrings- og hente-enhet (AS/R) mellom reolene. AS/R-enheten henter hyller fra reolene og midlertidig lagrer de i et plukkområde. Etter at plukking er utført, settes hyllen tilbake i reolen.

Målene med denne oppgaven er å undersøke påvirkningen på ordreplukkingstid når etterfylling og frasetting utføres parallelt med ordreplukking, og hvordan endringer i lagerpolicyene påvirker ordreplukktiden. Flere forskningsspørsmål vil bli besvart for å oppnå disse målene: *1)* Hva er aktuelle lagerpolicyer for et system med flere VLM-er som jobber i rekkefølge? *2)* Hva er innvirkningen på ordreplukkingstid i et system med flere VLM-er når etterfylling og frasetting utføres parallelt med ordreplukking? *3)* Hva er innvirkningen på ordreplukkingstiden når du bytter mellom aktuelle lagerpolicyer i et system der flere VLM-er opererer i rekkefølge? *4)* Hva er innvirkningen på ordreplukkingstiden når antall operatører og aktive VLM-er endres?

Den første delen av oppgaven er et teoretisk studie om hvordan et lager fungerer, dets funksjoner og aktiviteter, og ulike policyer for disse funksjonene og aktivitetene. Deretter følger en litteraturgjennomgang for å kartlegge den nåværende forskningen på VLM-er, der det er et tydelig gap i hvordan etterfylling og frasetting påvirker et system med VLM-er og hvordan flere VLM-er som opererer i rekkefølge blir utsatt for endringer i lagerpolicyene. Den tredje delen er en casestudie av et distribusjonssenter for ikke-farmasøytiske varer som består av flere VLM-er hvor etterfylling og frasetting utføres parallelt med ordreplukking. Deretter følger en del der aktuelle lagerpolicyer diskuteres. Den fjerde delen beskriver hvordan en Discrete Event Simulation (DES) modell er brukt for å undersøke effekten av å endre lagerpolicyene.

Hovedkonklusjonen i denne studien er at etterfylling og frasetting ikke påvirker ordreplukkingstiden eller ventetiden under ordreplukking. Oppholdstiden for en ordre økes derimot når etterfylling og frasetting utføres parallelt med ordreplukking. Videre påvirker lagerpolicyene plukktiden for ordre, enten positivt eller negativt, avhengig av policyen. Til slutt, når antallet operatører øker, øker utføringstiden for etterfyllingsordre, mens når antallet aktive VLM-er øker, reduseres ordreplukkingstiden.

Table of Contents

List of Figures

List of Tables

Acronyms

AS/R Automatic storage and retrieval.

CBS Class based storage.

DES Discrete Event Simulation.

EOQ Economic order quantity.

FEFO First expire first out.

FIFO First in first out.

FOI Fixed order interval.

HMN Helse-Midt Norge.

I/O Input/Output.

ICT Information and communication technology.

LS HMN Logistikksenter Helse Midt-Norge.

MAPE Mean Absolute Percentage Error.

NTNU Norwegian University of Science and Technology.

OOS Order oriented slotting.

PIS Perpetual inventory system.

ROP Reorder point.

RQ Research question.

SKU Stock keeping unit.

SLR Systematic literature review.

VBA Visual Basic for Applications.

VLM Vertical lift module.

WMS Warehouse management system.

Chapter 1

Introduction

1.1 Background

Warehouses are widely used within supply chains to store items for various lengths of time (Kay 2015). They are used to match customer demand, reduce transportation costs and provide better customer service (Hackman 2019; Stevenson 2015). A warehouse ensures that the correct items are available at the right location at the right time, and it enables items to be collected, sorted and distributed efficiently (Kay 2015). Further, a warehouse has multiple functions and activities including replenishment, put-away, storage and order picking. Replenishment is the function of restocking the storage, and it determines when and how much to restock. Put-away is the activity of moving the item into its correct storage location. Storage is when the items are waiting to be further processed or picked. Lastly, order picking is the activity of retrieving items from storage based on customer requests or customer orders. Order picking is the most cost- and time-consuming activity of a warehouse (de Koster et al. 2007; Kay 2015).

Order picking within a warehouse with manual operations requires an operator who travels between storage locations to retrieve the goods specified on the customer order. A storage location can consist of a single pallet with multiple cases, a case with numerous pieces within or single pieces. These are known as the three levels of order picking (Kay 2015), see Figure 1.1. As demonstrated by de Koster et al. (2007), up to 55% of warehouse operating costs are related to order picking when an operator is used. Further, up to 50 % of the order picking costs are related to travelling (Tompkins 2010), which means up to a total of 25% of a warehouse's costs are solely associated with travelling during order picking. These costs can be reduced by optimising the material flow, the manual processes, through heavy investments in automated solutions, or a combination.

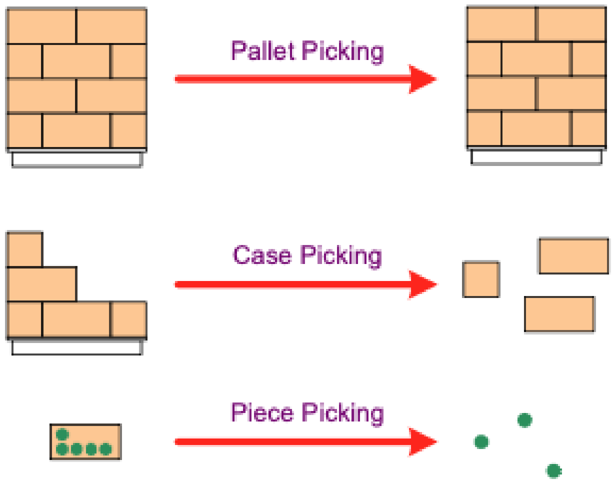


Figure 1.1: The three main levels of order picking
Kay (2015, p.33)

The operators can manually carry out warehouse functions and activities, or the functions and activities can be automated through heavy investments. de Koster et al. (2007) classifies the different manual and automated systems, depicted in Figure 1.2. Manual operations can be distinguished into three main classifications:

- **Picker-to-parts** is a system where the items are static, and the operator moves around the warehouse to locate and pick the right items.
- **Put system** is a process of retrieving multiple items and then distributing these to the different customer orders.
- **Parts-to-picker** is a system where the items are routed towards a static operator.

Parts-to-picker combines employing humans and machines, where the machines carry out the material handling before the operator executes the picking activity. MHI (2023a) Defines material handling as *"the movement, protection, storage and control of materials and products throughout manufacturing, warehousing, distribution, consumption and disposal"*.

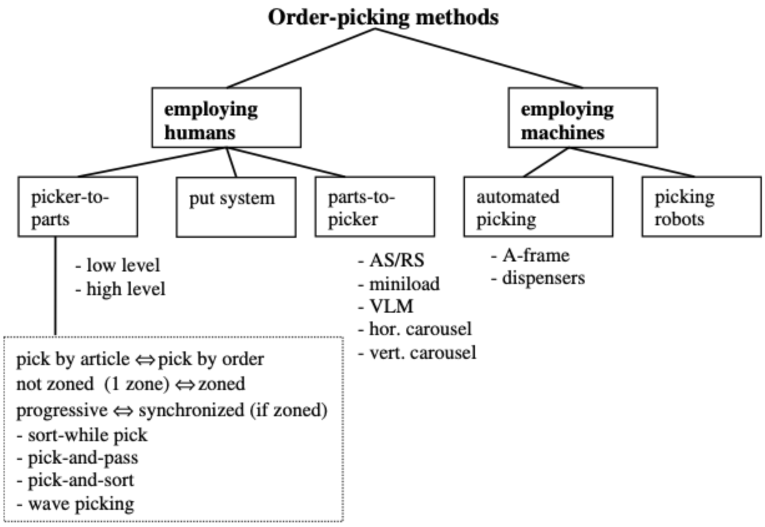


Figure 1.2: Classification of order picking systems
de Koster et al. (2007)

Order picking of single-case units or piece-picking, see Figure 1.1, can be more labour intensive than pallet picking due to increased order complexity and increased need for travelling between the storage locations (Hackman 2019). The most labour-intensive order picking is within a system with single-case picker-to-parts. Here, the operator picks a single case at a given storage location and then travels to the following storage location to pick another item.

The items that are eligible to be picked are usually stored at the lowest levels of a rack or on the floor, known as *forward pick storage*, while the rest of the levels in the rack are used for *reserve storage* (Kay 2015). Usually, only one item is available in a forward-picking storage slot and when this slot is empty, it is replenished from the reserve storage. The replenished item can be the same or a new item, depending on how the warehouse is managed. Further, since the forward-picking storage slot only stores one type of item, the demand for forward-picking slots increases when the variety of items increases. Simultaneously, the space needed for these picking slots increases, increasing the distance for order pickers.

One way to minimise the total order picking time and increase the warehouse throughput is to reduce the travelling needed to pick an order (Battini et al. 2015). This can be achieved in multiple ways, where one solution is to dedicate areas for single-case or single-piece picking. Here, various items are put into the same forward picking storage area where all items are within the operator's reach. Another solution is introducing new storage systems and technology that help ease the picking activity (Sgarbossa et al. 2017). One such technology is the Vertical lift module (VLM), see Figure 1.3.

Figure 1.3: A graphical illustration of a Vertical lift module
MHI (2023b)

The VLM is an enclosed pair of racks where trays are stored within, and the items are stored on the trays. An Automatic storage and retrieval (AS/R) device is in between the racks, which stores and retrieves the trays. When an item is due to be picked, the tray with the item is retrieved and displayed in a picking bay. The operator can pick the item corresponding to the order from the picking bay. When the item is picked, the tray is retrieved and stored in one of the racks. A VLM is a parts-to-picker system where travelled distance is reduced to a minimum and storage volume is utilised (de Koster et al. 2007). Further, the VLM is typically throughput constrained and not storage-constrained (Meller and Klote 2004). A VLM can have multiple configurations, where the single bay and double bay are the most mentioned in the literature, as presented in Section 3.2. A double bay VLM has two picking bays to display trays for the operator. To further utilise the benefits from VLMs and to optimise the picking time, the VLMs are often clustered in groups called pods. Inside one pod, the operator can pick from multiple VLMs simultaneously, reducing the travelling needed (Meller and Klote 2004). Furthermore, a pod configuration enables the operator to pick a broader range of products because more forward pick storage locations are available nearby.

The picking time can be further reduced by finding the optimal way of managing the orders. For example, the orders can be controlled by splitting the pick area into zones or batch multiple orders together (Nicolas et al. 2018). Other means of reducing order picking time are finding the optimal storage and replenishment policies. These determine how the products should be stored and how they should be refilled. When using VLMs, the distance from the bay and the required tray will affect the tray retrieval time. The further away the tray is, the longer its retrieval takes (Daria et al. 2015; Mantel et al. 2007; Sgarbossa et al. 2019). Finding the optimal storage policy will enable the retrieval to be executed more rapidly. de Koster et al. (2007) assumes that the replenishment and put-away are done off-shifts or when the VLMs would not be used for order picking. However, replenishment and put-away are vital to ensure available products are in storage (Hackman 2019). The most optimal policies must be determined for a system where replenishment and put-away must be executed parallel with order picking.

Previous studies on VLMs are mainly focused on the throughput of a single VLM under different configurations, and only a few studies have examined the effect of multiple VLMs. Meller and Klote (2004) designed the first throughput model of VLM pods, Nicolas et al. (2018) investigated the effects of order batching, and Calzavara et al. (2019) carried out an economic evaluation of when to use VLMs and how many to acquire. Also, all studies have excluded the replenishment activity for their throughput simulations or calculations based upon the assumption made by de Koster et al. (2007), even though replenishment is vital for a warehouse to function properly (Hackman 2019).

As a cost-reducing measure, Helse-Midt Norge (HMN) has decided to replace their local warehouses with a new regional logistics centre, known as Logistikksenter Helse Midt-Norge (LS HMN). HMN is the governmental-owned healthcare provider for the middle part of Norway, and their responsibilities are to provide healthcare to patients, educate healthcare personnel and research (Norge 2022). HMN operates all their facilities, including their warehouses and distribution channels. Their old warehouse facilities relied on manual operations and activities, with little to no support of Information and communication technology (ICT), while the new facility is modernised. In the new warehouse, LS HMN, many warehouse functions are automated and digitalised through heavy investments in new technology. As a result, picking and replenishment activities will differ in technological solutions and require new thinking regarding planning, project placing and task sequencing. LS HMN will consist of multiple VLMs used as a low-level parts-to-picker system with human order picking and replenishment.

Further, LS HMN have multiple VLMs operating in sequence. While item N in an order is picked, the VLM with item N+1 can fetch and display the tray with item N+1. This logic implies that two VLMs are active simultaneously for one order. The amount of active VLMs per order is decided by the *active VLMs parameter*. The active VLMs parameter is defined as an integer, and it decides how many consecutive items in an order the warehouse should make available for picking simultaneously.

1.2 Research Questions and Objectives

The overall scientific goals of this thesis are to find out how the replenishment and put-away activities affect the order picking time when they are executed in parallel with order picking. Further, how is the order picking time affected when specific warehouse policies are changed in a system where multiple VLMs operate in sequence. To achieve these goals, the following Research question (RQ)s will be answered:

RQ1 What are the applicable warehouse policies for a system with multiple VLMs working in sequence?

This RQ first aims to clearly define how a warehouse operates and how the different warehouse functions and activities are executed under different policies. Further, it is discussed how these policies can be applied to a warehouse with multiple VLMs. This question will be answered through a literature study on warehouse activities, functions, and corresponding policies, which will be discussed based on a case study.

RQ2 What is the impact on order picking time in a system with multiple VLMs when replenishment and put-away are executed in parallel with order picking?

This RQ aims to examine how the replenishment and put-away function and activity affect the order picking time. The question will be answered using a simulation model to compare scenarios with and without parallel replenishment and put-away.

RQ3 What is the impact on the order picking time when changing between the applicable warehouse policies in a system where multiple VLMs operate in sequence?

This RQ aims to examine how different warehouse policies affect the order picking time in a system where multiple VLMs operate in sequence. The goal of this RQ is to find which combination of the different applicable warehouse policies, as discussed and answered in RQ1, enables, on average, the fastest order picking time. This RQ will be answered using a simulation model where scenarios with different policy combinations will be compared.

RQ4 What is the impact on the order picking time when the number of operators and active VLMs are altered?

When multiple VLMs operate in sequence, multiple VLMs can serve the same order simultaneously. The active VLMs parameter decides how many consecutive items the system should prepare simultaneously for one order. Further, the number of operators can be altered simultaneously with the changes in the active VLMs parameter. The impact of these changes will be answered using a simulation model where the parameter and the number of operators are changed.

1.3 Research scope

The scope of this research is confined to warehousing and its functions and activities, explained in detail by Kay (2015) and Hackman (2019). The basics of a warehouse are investigated based on the typical warehouse functions and activities listed by Kay (2015). These functions and activities include replenishment, put-away, storing and order picking, and all are executed within the LS HMN. Further, the focus is on the policies applicable to these four functions and activities. The case study limits the scope to the specific technology applied at LS HMN, the VLMs. Lastly, the combination of theoretical aspects of a warehouse, policies regarding the four functions and activities, and the case study description provides the complete scope. Figure 1.4 shows the relationship between the theoretical aspects and the case study.

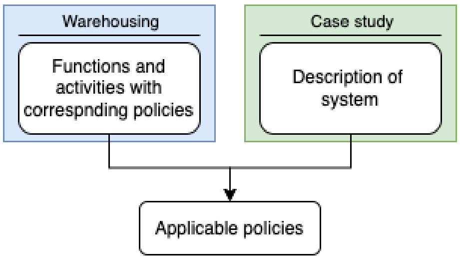


Figure 1.4: Thesis scope relationship

1.4 Thesis structure

The thesis is structured based on multiple motivational aspects. Firstly, LS HMN have invested in a new warehouse which uses multiple VLMs that operate in sequence. Secondly, there is little research on how multiple VLMs operate and how they react when subjected to changes regarding warehouse policies. Lastly, a research gap exists regarding replenishment executed in parallel with order picking. These three motivational aspects are seen in the first part of Figure 1.5. All these motivational aspects lead to multiple research questions, seen in the second part of Figure 1.5. The methodology needed to answer these questions and the outcome of the questions is shown in the third part of the figure. The first part of chapter 3 is a literature review stating the functions and activities of a warehouse and stating multiple policies on how these functions and activities can be executed. The second part follows a literature review needed to show the current state of research on VLMs.

After the literature reviews follows a case study. Firstly, the chapter outlines the system at LS HMN with a conceptual drawing of their layout and material flow. Further, the chapter details how the VLMs are used in this specific case, together with their warehouse policies. In the second part of chapter 4, the warehouse policies uncovered in chapter 3 are discussed based on how they could fit in a system with multiple VLMs operating in sequence. The disussion is based on the first part of chapter 4 and the results of the discussion is the answer of RQ1. Further, in chapter 5, a simulation model of LS HMN is built, and its logic, functionality, and the scenarios run are explained in detail. The result of the simulations is shown in chapter 6. The results are then discussed in chapter 7. The outcome of the discussion is the answer of the rest of the RQs.

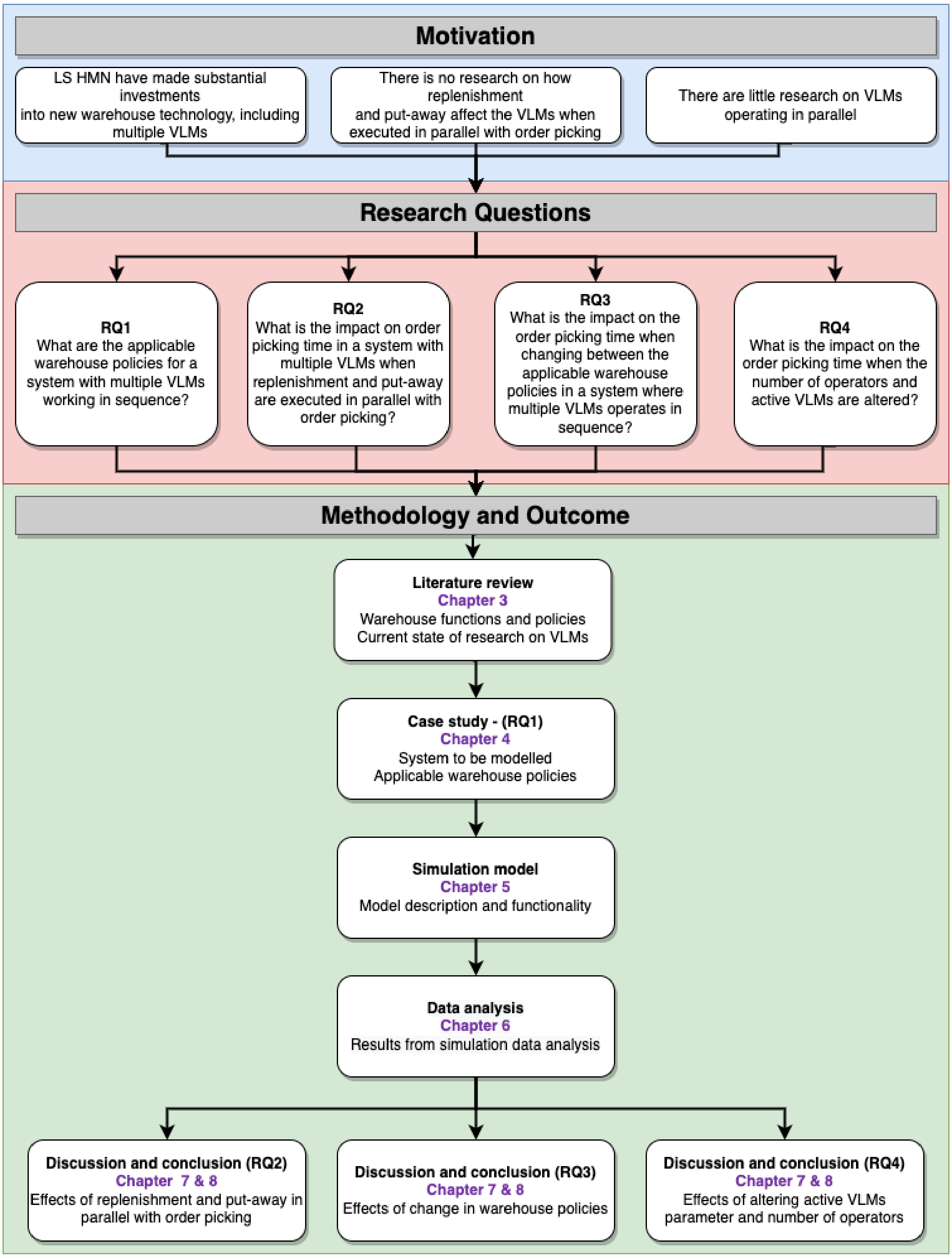


Figure 1.5: Thesis structure

Chapter 2

Methodology

In this chapter, the different methods used to answer the RQs are described and motivated. Firstly, the literature review concept is described before describing how it is used in this thesis. Then follows a description of a case study and how it has been carried out. Finally, the last section motivates why simulation was chosen as a method and how data regarding the simulation have been processed.

Some of the content in this chapter is directly cited from an unpublished report written as the final assessment in the course *TPK4530* given at the Norwegian University of Science and Technology (NTNU). The author participated in this course during the autumn of 2022.

2.1 Literature study

Two separate literature reviews have been conducted in this study, the first for the theoretical background of the thesis and the other to highlight research gaps in the topic of VLM functions, operations and management. First, a literature review is used to identify theories and previous research on specific topics (Ridley 2012). Further, the literature review could be used to identify the problems of the thesis or highlight research gaps which should be filled. For this thesis, the literature reviews lay the basis for the theoretical aspects, presented in chapter 3, and the research gaps on the VLM topic, presented in Section 3.2. These are some of the multiple purposes of a literature review, as stated by Ridley (2012). Other purposes of a literature review could be to provide the historical background of the current topic or to introduce relevant terminology and definitions.

The first literature review aims to uncover the theoretical aspects of warehousing and its functions and activities with related policies. These have been found through literature searches in *Scopus* and *Google Scholar*, as well as through multiple textbooks.

Specific keywords used for the literature searches conducted in the databases are listed in Table 2.1.

Table 2.1: Keywords used for the literature study on warehousing

Warehousing operations	Warehousing functions
Warehousing policies	Replenishment*
Put-away*	Storage*
Order picking*	Vertical lift module / VLM

*Searched by itself and with warehousing or policy/policies, or a combination

The literature review on the topic of VLM is based on a Systematic literature review (SLR). Tranfield et al. (2003) describes a SLR as a replicable, scientific and transparent process. It is a structured way of finding literature related to the specific topic, which is easy to repeat and with a clear line of action. The method enables other researchers to validate the results and to re-do the same literature review. The method contains multiple stages, which are characterised and described by Tranfield et al. (2003):

- **Stage 1 - Planning the review:** Establish a review panel which defines a review protocol. The protocol contains information regarding specific questions of a given field, search strategy for the review (e.g. databases to search within, specific keywords) and criteria for excluding or including search results.
- **Stage 2 - Conducting a review:** Carry out the literature search by following the review protocol and search strategy, and report the findings in a structured way. Evaluate the full list that respects the inclusion and exclusion criteria. The last steps are to screen the literature on title and abstract before a full-text reading and analysis are performed.
- **Stage 3 - Reporting and dissemination:** Present the results by providing a complete descriptive analysis of the given field and summarise the findings for multiple categories. The categories are either found by a deductive approach, constructing the categories based on previous studies or existing theories, or by an inductive approach, categories are chosen based on the material under examination.

The keywords used for the literature review on the topic of VLM are seen on the far left in Figure 2.1. Category 1 specifies that the given technology, VLM, has to be included in either the paper's title, abstract or keywords. Further, Category 2 narrows the search results to give literature for managerial aspects, and not technical aspects for the use of VLMs. The search gave a total of 513 articles and papers, and most of the papers were out of scope. The 513 documents were all exported into a spreadsheet and systematically reviewed. Firstly, the documents were screened based on their title. In this step, most documents were excluded because they were out of scope. VLM is an abbreviation used in fields other than warehousing, which explains the number of papers out of scope.

In the second step, all the abstracts were read. Unfortunately, multiple papers were out of scope, and some documents were non-scientific articles that got excluded. In the end, after screening titles and abstracts, 11 papers were comprehensively analysed. The final results had documents that researched VLMs based on different variable factors and evaluation criteria. The complete procedure is shown in Figure 2.1, and the results are presented in Section 3.2. The search was done on the 19th of September 2022 through the *Scopus* database.

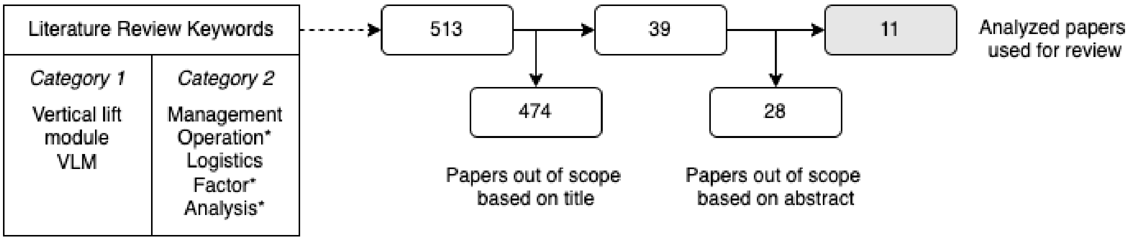


Figure 2.1: Literature review process

2.2 Case study

A case study is research based on a limited number of cases, and the analyses of these (Voss et al. 2002). Only limited statistical analysis can be applied to case studies because of the limited number of cases. Further, a case study is time-consuming and drawing generalised conclusions from only a limited set of cases requires care. However, a case study can have a very high impact. Voss et al. (2002) states that when unconstrained from the limits of questionnaires and models, the case study can lead to new and creative insights and the development of new theories.

Further, the case method enables why, what and how questions to be answered, as well as studying a system during its natural setting (Voss et al. 2002). For case studies, there is no defined amount of cases needed. If one case is analysed, the case can go more in-depth, but drawing generalised conclusions, models, or theory is more limited. Contrary, if multiple cases are used, the depth of the study might be reduced, but drawing generalised conclusions from multiple studies are more credible (Voss et al. 2002).

For the case study within this thesis, only one case is analysed in-depth. The case company is chosen because they apply a specific warehouse technology, the VLM. Other than the VLMs, the warehouse operations are described to give an overview of the warehouse, but the main focus is the VLMs and how they operate. The specific case is used for a system which can be simulated and for input data needed for simulating. The specific type of warehouse, other technologies they use and the types of items they distribute are irrelevant to the case but are mentioned to give a holistic view of their operations.

2.3 Discrete Event Simulation

Multiple approaches exist for studying a system and answering *What if* questions. Firstly, one can experiment with the existing system or with a model of the system. For some systems, there might be a feasible solution to experiment with the existing system if it is possible and cost-effective, but altering systems might be physically impossible or too costly (Law 2015). Therefore, it can be more reasonable to experiment with a model of a system. This can be done by making a physical model or a mathematical model. A physical model could be a tabletop model of a system or miniature boats floating in a pool. However, making a physical model for operations research or systems analysis is difficult and expensive, which is why mathematical models exist. Mathematical models represent systems in terms of logical and quantitative relationships that are easily manipulated (Law 2015).

Further, one can either have an analytical solution or a simulation. An analytical model can be used if the model is simple enough to get an exact analytical solution. Analytical solutions can be simple enough to be calculated with pen and paper, while the more complex the system is, the more computational power is needed. When the model is highly complex, subject to variability or interconnectedness, a simulation model is a preferred solution (Robinson 2004). Law (2015, p.5) defines a simulation as "*numerically exercising the model for the inputs in question to see how they affect the output measures of performance*. Figure 2.2 summarises the different ways to study a system.

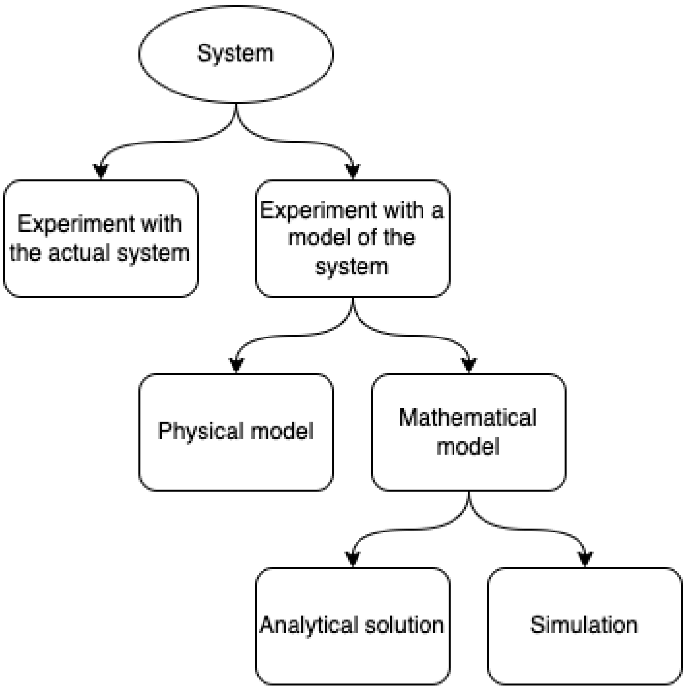


Figure 2.2: Ways to study a system
Law (2015, p.4)

Simulation, compared to experimenting with the real system, has multiple advantages (Robinson 2004):

- It can be cost-efficient because there is no need to interrupt the ongoing operations to try out new ideas or to alter the system. The daily operations can still continue in parallel to the simulation.
- It is not as time-consuming as experimenting with the actual system. When altering the actual system, it might take weeks or months to show the results, while in the simulation, it might only take a few minutes.
- Control of the experimental conditions.
- The real system might not even exist.

Despite these advantages, there are also some disadvantages with simulation. Robinson (2004) lists multiple disadvantages of simulation:

- Simulation software is expensive, and the hours needed to develop a model can be costly.
- Developing a simulation model is time-consuming.
- A simulation needs a lot of input data. For example the geometry of objects, customer orders or processing time for items. It is not given that the data required exists, but if it does, it might have to be manipulated to be suitable for the simulation.
- When interpreting the results from a simulation, they might not be as realistic as hoped. The validity of the simulation, the assumptions and the simplifications all have to be considered.

Discrete Event Simulation (DES) is a simulation method where only the points in time at which the state of the system changes are represented (Robinson 2004). The system is modelled as a series of events when a state change occurs, based on time (Law 2015; Robinson 2004). Examples of such events are an operator starting a machine, a customer arriving at a desk or an operator picking an item.

2.3.1 The simulation software

This study uses a DES model to manipulate a real system to answer multiple *what if* questions. The software used is Flexsim®. Flexsim® is a 3D simulation modelling and analysis software that helps to understand and improve any system or process

(FlexSim® 2023a). The software enables the development of a DES model with complex logic in a 3D environment. The software can simulate multiple industries, including manufacturing, warehousing and material handling, supply chains and healthcare (hospitals). Figure 2.3 shows an example of a complex material handling system modelled in Flexsim®.

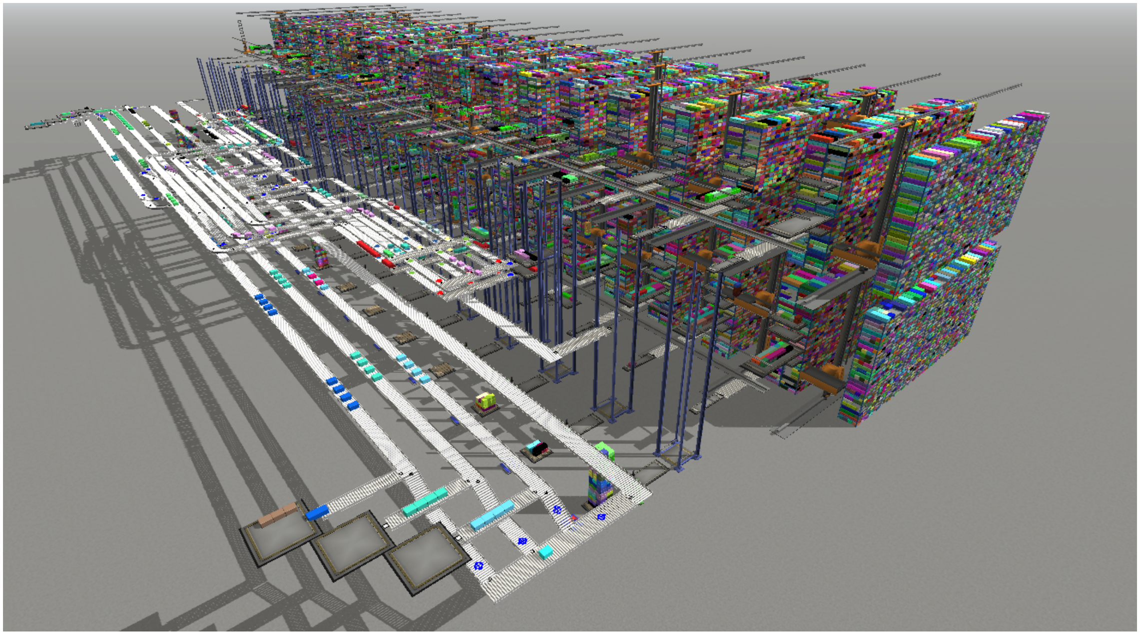


Figure 2.3: An illustrations of the 3D-environment of a DES model created in Flexsim® Blikås et al. 2021

The software is split into two main parts, the 3D environment and a Process Flow tool. The 3D environment shows a model of the system under study with all its corresponding geometry, connections and animations. The 3D model consists of fixed resources, such as racks or processors, task executors, such as operators or cranes, and flow items. A flow item represents an item that moves and interacts with the fixed resources. These could be pallets, boxes or any item which should be altered. The 3D environment could be used to make basic logic within the model but it is mainly used to visualise the model.

The Process Flow is used to make complex logic within the model. The Process Flow is an environment with drag-and-drop usability where activities and resources are sequenced. Tokens are sent from one activity to another, where data stored on the token is modified. The Process Flow could be linked to the 3D environment, where the Process Flow determines the logic, and the 3D environment visualises the events that happen. A token within the Process Flow can represent a task that a task executor must do, a flow item within the 3D environment, a customer order or a combination. Figure 2.4 shows a simple process flow. The green circles are tokens, the activities are the light blue blocks and the resource is the beige.

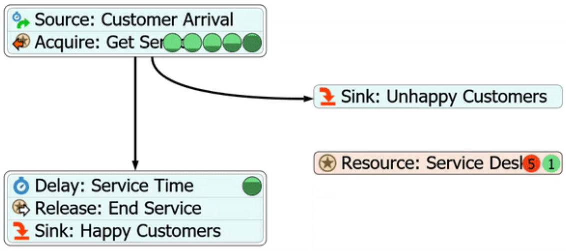


Figure 2.4: An example of a basic process flow made in Flexsim®
FlexSim® 2023c

Further, Flexsim® has a tool called the *Experimenter* which automates running multiple scenarios (FlexSim® 2023d). The experimenter allows the user to define multiple scenarios based on model inputs. These inputs can alter the model, e.g. the number of operators or which warehouse policy to use. After all the scenarios are defined, multiple simulations are run simultaneously, one for each scenario. The experimenter also enables the possibility to run multiple replications of each scenario. The results of each scenario, and replication, can be viewed and explored in a performance measures window if desired. Within this thesis, the experimenter has been used to reduce the time needed to run the scenarios by running multiple simultaneously. Only one replication has been run for each scenario.

2.3.2 Data processing

In-house routines were made to import, export and analyse data. The imported data were given from LS HMN and restructured to fit the simulation model. Multiple data sets were exported from the model to verify and validate the model and data to analyse the different scenarios. Verification and validation are whether the model and its results are correct and valid for its specific purpose (Sargent 2020). Verification of a model ensures that the model achieves what it is supposed to do without any errors, and validation is to ensure the model's accuracy (Kleijnen 1995). Better accuracy leads to more realistic outcomes of the model, but having a super realistic model is time-consuming and expensive. Therefore, the model's accuracy should not exceed what is required for its purpose (Sargent 2020). A detailed explanation of imported and exported data can be seen in Section 5.2, and the steps taken to verify and validate the model are listed in Section 5.5. Further, two Visual Basic for Applications (VBA) scripts were written to import and analyse the extracted data into Excel. The first script was used to import and restructure the data for easier access. A total of 392 (98 scenarios with 4 data sheets each) data sheets with varying amounts of data were imported and restructured. The second script was written to analyse and calculate the 392 data sheets.

2.3.3 Regression analysis

Regression analysis is often used to test and describe causality between variables. However, a regression analysis cannot prove relationships between factors and variables. It is only possible to test if the relationship is statistically significant from zero. Therefore, the analysis is used to find casual relationships, and it is used for hypothesis testing. Further, regression is often used to see how changes in independent variables explain changes in the dependent variable. The independent variable is where the impact of changes wants to be measured. While the dependent is the variable that is presumed to be affected by changes in the independent. (Oppen et al. 2020)

Further, regression analyses can be executed with one independent variable, known as a bivariate regression, or with multiple independent variables, known as a multivariate regression. Within this thesis, multivariate regressions are used to examine if there is a statistical relationship between the dependent variable and the independent variables. When executing a regression analysis, the statistical significance has to be checked. The statistical significance describes the probability that our results are random. Further in this thesis, the statistical significance will be referred to as *P-value (sig.)* . The lower the P-value, the lower the possibility that the results are random. For this thesis, a P-value between 0.10 and 0.05 shows a weak statistical significance but affects the dependent value. A P-value between 0.05 and 0.01 shows a statistical significance, while a P-value <0.01 shows a strong statistical significance. (Oppen et al. 2020)

Chapter 3

Literature Review on Warehouse Policies and VLMs

The following chapter includes the theoretical aspects needed in order to explain the rest of the thesis. First, the basic theoretical aspects of warehousing are introduced, which leads to the examination of multiple warehouse policies. After the theoretical aspects have been defined, a literature review on previous research of VLMs is presented. Multiple papers on the topic of VLMs are analysed and the results are summarized at the end. This chapter is motivated by the lack of research on VLMs, as shown in Figure 1.5.

Most of the content in this chapter is directly cited from an unpublished report written as the final assessment in the course *TPK4530* given at the NTNU. The author participated in this course during the autumn of 2022.

3.1 Warehousing functions, activities and related policies

Warehouses are widely used within supply chains to store items for various lengths of time (Kay 2015). They are used to match customer demand, reduce transportation costs and provide better customer service (Hackman 2019; Stevenson 2015). A warehouse ensures that the correct items are available at the right time, enabling the items to be collected, sorted and distributed efficiently (Kay 2015). Figure 3.1 illustrates the benefits of transportation with a warehouse in the supply chain. With this example, having a warehouse reduces the number of routes and transportation costs by one-third while still keeping up with the demand.

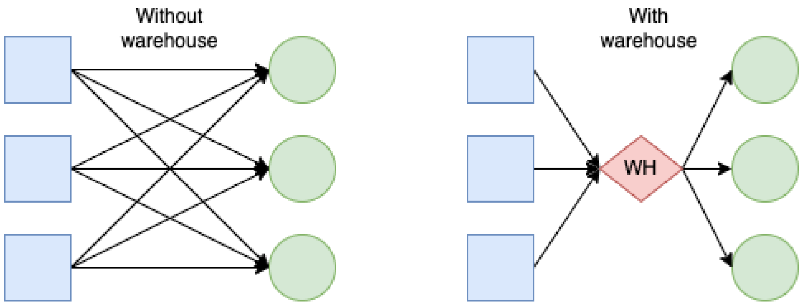


Figure 3.1: The difference in transportation with and without a warehouse. By having a warehouse, one can reduce travelling significantly. Each line refers to one route.
Hackman (2019, p.7)

A warehouse consists of multiple functions and activities. First, the items are received, then cross-docked, put into reserve storage or put straight to forward picking storage. Cross-docking is when a customer has already ordered an arriving item, so the item can be shipped immediately, eliminating the need for storage (Hackman 2019). If the received item is not cross-docked, it is put into reserved or forward picking storage. As mentioned earlier, forward picking storage is where items eligible for picking are stored until it is picked. The forward picking storage makes the items more accessible, enabling more efficient order picking (Kay 2015). Items are stored in the reserved storage until the forward picking storage needs replenishment or until a customer orders the item. The item is then picked and sent to a packing, sorting and unitizing area or shipped directly. Put-away is moving items from one location and placing them in the correct new location (Frazelle 2016). Figure 3.2 depicts the material flow for a typical warehouse.

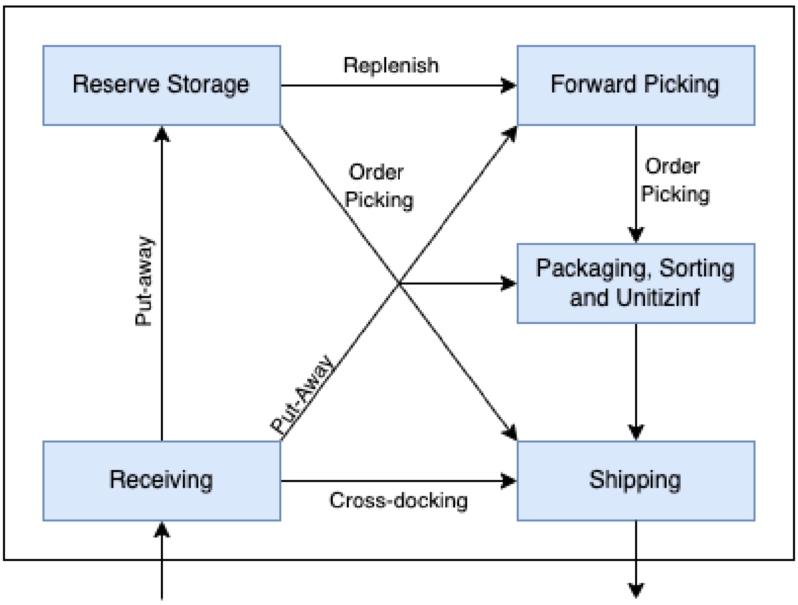


Figure 3.2: Typical warehousing functions and activities.
Kay (2015, p.24)

A Warehouse management system (WMS) is needed to coordinate the functions and activities within a warehouse. A WMS is a complex Information and communication technology (ICT) system which takes hold of all information within the warehouse in a paperless fashion (Kay 2015). The WMS ensures coordination between all processes, including inventory management, storage locations, orders and workforce (Hackman 2019). It knows everything about the items, from size and weight to packaging and storage locations. The software receives the orders and transforms them into picking lists. A picking list is a sequence of Stock keeping unit (SKU)s and its corresponding pick amount, as well as the location of the given SKU. The picking list provides the operator with a sequence to ensure the order is picked correctly (Kay 2015).

All the functions and activities within a warehouse can be executed in multiple ways, called policies. Furthermore, the policies require different degrees of data management, ICT, coordination and operator assessment to be executed. Based on the scope of this thesis, only four warehouse functions and activities will be explained in detail; replenishment, put-away, storage and order picking.

3.1.1 Replenishment

Replenishment is when and how much to order to fulfil customer demand and satisfaction and to reduce the risk of stock-out. In order to know when to replenish, the inventory must be monitored. Stevenson (2015) lists two main strategies of monitoring inventory, a periodic system and a Perpetual inventory system (PIS). In a periodic system, the stock is counted within fixed time intervals. This strategy can potentially reduce ordering costs because multiple products are ordered simultaneously. However, it limits the inventory control between reviews and increases the risk of stock out between restocks.

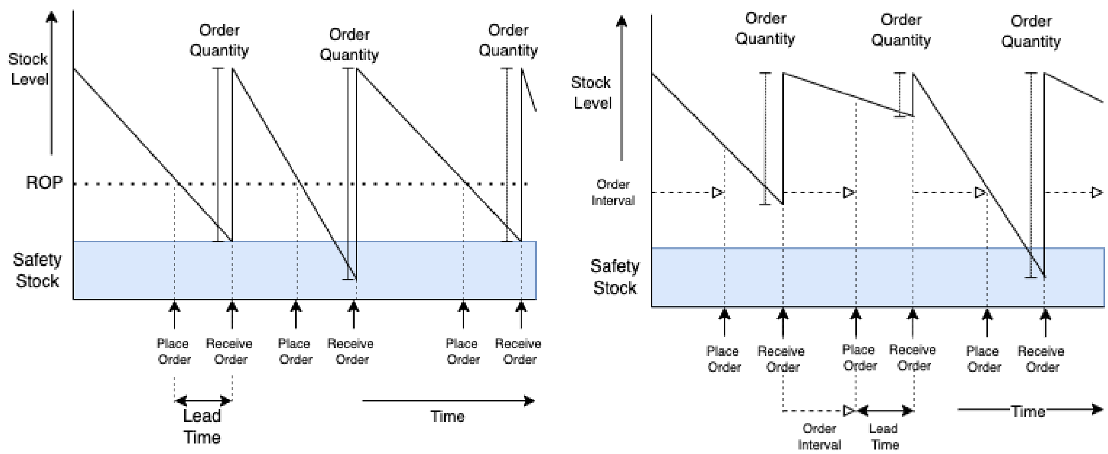
On the other hand, the PIS keeps track of the inventory continuously. When an item is removed from storage, the action is recorded, and the stock is updated. When the stock reaches a certain point, an order with a predefined amount can be placed. With this method, an optimum order quantity can be determined. In other words, in a periodic system, a physical stock count is carried out on set time intervals, while in a PIS, the stock is updated when an item is removed from inventory. There are multiple different policies for replenishment (Stevenson 2015):

- **Two-bin system** - A simple PIS where two bins are used for inventory. When the first bin is empty, a replenishment order is placed while the second bin is used. The bins hold enough stock to fulfil demand during the lead time of the replenishment order and a buffer, referred to as safety stock. There is no need to keep track of every transaction with this system.

- **Economic order quantity (EOQ)** - EOQ is a fixed order quantity that reduces order- and holding costs. It is based on a fixed annual demand with an even distribution throughout the year, fixed costs per item, per order and holding cost, and a fixed lead time. The method does not take hold of uncertainties and quantity discounts.

- **Reorder point (ROP)** - A PIS that determines when to order a fixed amount. An order is placed when the stock reaches a certain level, as depicted in Figure 3.3a. ROP takes hold of actual demand, and the demand determines the time of order. Orders might be placed more frequently for products with a high turnover and less frequent orders for low runners. The amount to order is predefined and calculated based on the maximum inventory of the given SKU, demand during the lead time and safety stock.

- **Fixed order interval (FOI)** - This replenishment policy uses a fixed time interval (weekly, monthly, yearly) for placing orders, while the amount is variable, as depicted in Figure 3.3b. The amount is determined by the available and the maximum stock. This method is a periodic system and a PIS. A stock count is carried out for a periodic system, and orders are placed up to a maximum amount. In contrast, an automatic order can be placed for PIS since the system knows how much stock is available and the maximum capacity.



(a) ROP replenishment policy (b) FOI replenishment policy

Figure 3.3: Two replenishment policies
Stevenson (2015, p.574)

3.1.2 Put-Away

Put-away is moving a product from a receiving area to a reserve storage location or from reserve to forward storage, to match the product's and location's characteristics (Kay 2015; Lam et al. 2009). For low-level order picking, the put-away activity relates to moving products from reserve storage to the forward picking area, as shown in Figure 3.2. The activity is carried out based on replenishment orders from the picking locations. Frazelle (2016) describes two primary practices of put-away, depending on when they are executed:

- **Direct primary** is used when put-away to the forward picking storage is executed immediately after the item arrives at the warehouse. The put-away can only be executed if there is an opening and the item and location characteristics are matched.
- **Direct secondary put-away** is used when the items are stored in the reserve storage until the forward picking storage requests the item.

Both these methods can be utilised simultaneously, depending on when the pick location is available. Further, Frazelle (2016) lists three strategies for put-away:

- **Nondirected put-away** - Operators choose locations themselves without an aim to maximise storage density or operating productivity. E.g. nearest to the floor, nearest the break room or at the first available spot.
- **Directed put-away** - A WMS supported strategy where the location is chosen to maximise space utilisation and retrieval productivity and to ensure good product rotation.
- **Prioritised put-away** - Sequencing the put-away based on priority.

Directed and prioritised are usually supported by a WMS to optimise and prioritise the put-away. The WMS should sequence the put-away to minimise execution time and to respect deadlines (Frazelle 2016). Put-away can be executed in multiple ways, shown in Table 3.1. Discrete is the simplest one, where one operator replenishes one item at a time, while batch enables multiple products to be replenished simultaneously. Zone distributes the put-away activity into zones depending on where the products are stored. For zone put-away, one could execute the put-away discrete, one item at a time, or combine the zone and batch methods. The zone and batch method enables the operator to execute put-away for multiple products simultaneously within the operator's given zone.

Table 3.1: Put-away methods. (Frazelle 2016; Kay 2015)

Method	Operators per put-away	Put-aways per picker
Discrete	Single	Single
Zone	Multiple	Single
Batch	Single	Multiple
Zone-batch	Multiple	Multiple

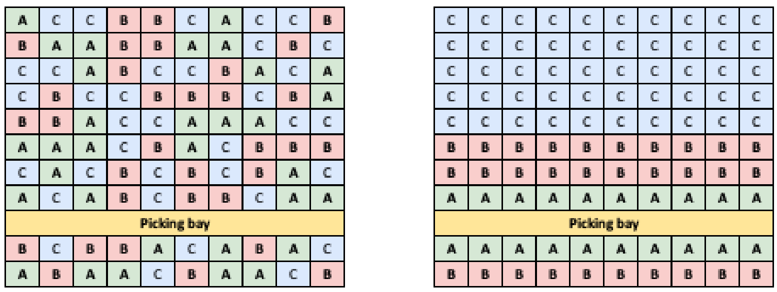
3.1.3 Storage

Both de Koster et al. (2007) and Kay (2015) discuss multiple storage policies. These rules decide where a product of a given SKU should be stored within the warehouse. According to de Koster et al. (2007), the following list is some of the most used policies:

- **Random storage** - All incoming pallets or items are given a random eligible empty location (de Koster et al. 2007). For storage systems that are managed by a WMS, the location is chosen with an equal probability (de Koster et al. 2007), while for a warehouse with manual operations with little to no support from an ICT-system, the closest open location is usually chosen (Kay 2015). The nearest open location will be random, leading to full storage locations close to the Input/Output (I/O) point, while locations further away will gradually have more available space (de Koster et al. 2007). The random storage policy results in high space utilization and minimizes building costs, but at the expense of increased travel distance and handling costs. Figure 3.4a illustrates random storage within a VLM.
- **Dedicated storage** - Each SKU has one or multiple fixed predefined storage location(s) and each SKU has to have enough locations to store the maximum inventory level (de Koster et al. 2007; Kay 2015). This policy minimizes handling costs because pickers get familiar with product locations, but it maximizes building cost and space needed. Another disadvantage of this policy is that when a location is empty, it is still reserved for an item that might be out of stock.
- **Full turnover** - Distributes the storage location of the SKUs based on their turnover. Products with the highest turnover rates are located closest to the I/O point and in the easiest accessible positions. The other products are then distributed further away according to their turnover rate. SKUs with the least turnover is stored the furthest away from I/O point. The policy enables quicker picking times for items with a high turnover. However, a disadvantage is that the demand constantly fluctuates, and so the SKUs will be assigned a new location frequently (de Koster et al. 2007). The reassignment of locations is constantly updated and executed by the VLM.
- **Class based storage (CBS)** - CBS combines random storage, dedicated storage and full turnover policies, where the SKUs are assigned to different classes based

on their turnover, and each class has its dedicated area. Within each class, the storage is randomized (Kay 2015). ABC classification, or the Pareto principle, is often used to establish the different classes (de Koster et al. 2007). The principle is to group the SKUs with the highest turnover and highest demand into the A-class. This is supposed to be 20% of the SKUs, which contributes to around 80% of the total amount of products picked. The B-class is for items with the second fastest turnovers, and the C-class is for the items with the lowest turnovers. A CBS assignment for VLM is depicted in Figure 3.4b.

- **Family grouping** - A storage policy where the relation between products is the main principle. Products often ordered together are grouped and stored in the same zone (de Koster et al. 2007). Mantel et al. (2007) developed a method called *Order oriented slotting (OOS)* which minimizes the total travelling time for VLMs by grouping products that are frequently ordered together in the same tray.



(a) Random Storage (b) CBS per tray

Figure 3.4: Two storage assignment policies applied to VLM
Daria et al. (2015)

3.1.4 Order Picking

Order picking is when materials or items are located and removed from storage to fulfil a specific customer order (Kay 2015). As mentioned earlier, it is one of the most time- and cost-consuming activities for warehouse operation costs (Tompkins 2010). Order picking consists of three phases: (Hackman 2019).

1. **Travelling** to the correct area or zone where the given product is stored. This is potentially the most costly and timely activity of order picking.
2. **Searching** for the correct item in the given zone. This is where the operator stands in front of multiple storage locations and has to locate the correct items. High volume and a large range of SKUs in the same area can make this a difficult and timely task, and picking errors can occur.

3. **Reach, grab and put** is the final step, and the value-adding step (Hackman 2019). Here the operator grabs the correct item and puts it in the corresponding tote of the order.

There are three levels of order picking, from pallet to piece picking, as seen in Figure 1.1. Pallet picking is when full pallets are picked, case picking is when cartons of items are picked, and piece picking is when individual units of the given item are picked (Kay 2015, p.33).

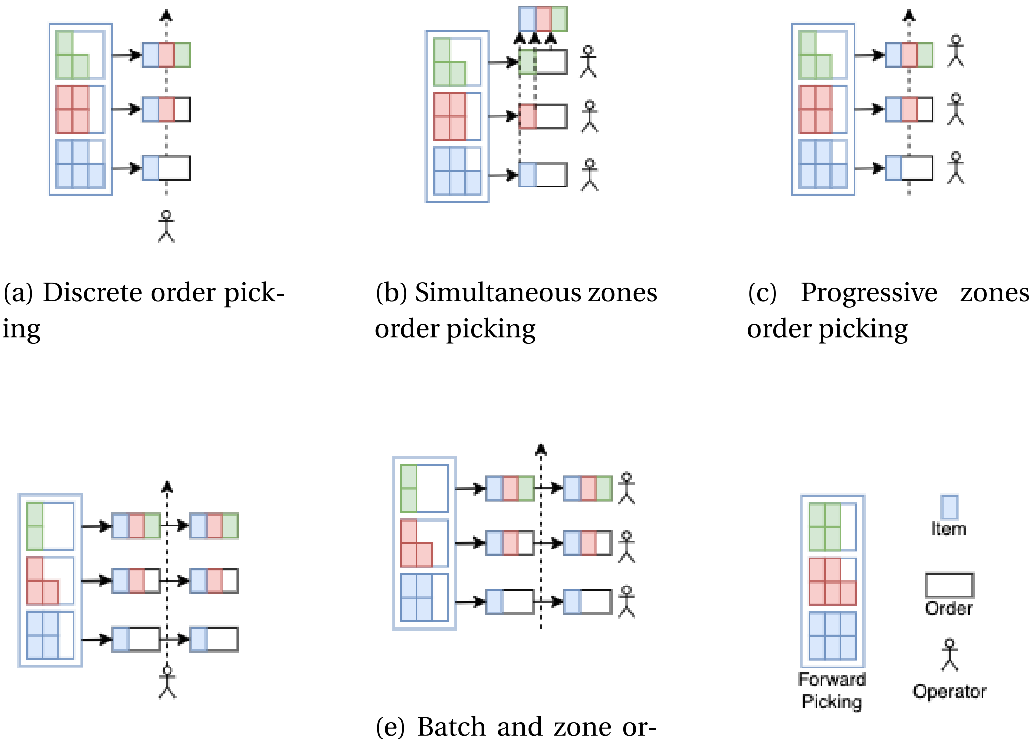
There are multiple methods of order picking which determine the number of pickers per order and orders per picker, shown in Table 3.2. Discrete picking is used when one picker picks a single order's items, depicted in Figure 3.5a. Zone picking is when a picker only picks items in the picker's given zone. There are two possible configurations of zone picking, simultaneous picking and progressive assembly (Kay 2015):

- In simultaneous picking, depicted in Figure 3.5b, the orders are split into different zones, and all zones pick simultaneously, hence the name. When all the zones are finished, the order is consolidated. This method minimizes picking time because all zones pick simultaneously.
- Progressive assembly, depicted in Figure 3.5c, is when one zone picks all the given items and then hands the order to the next zone. This configuration increases the total pick time but neglects the need for consolidation.

Batch picking is when a single picker picks multiple orders, depicted in Figure 3.5d. This method may reduce travel and search time if the products in the batch of orders are in the same location. The zone-batch method, depicted in Figure 3.5e, combines zone and batch picking where the orders are split into zones where order pickers can pick multiple orders simultaneously (Kay 2015).

Table 3.2: Order picking methods. Kay 2015, p.32

Method	**Pickers per order**	**Orders per picker**
Discrete	Single	Single
Zone	Multiple	Single
Batch	Single	Multiple
Zone-batch	Multiple	Multiple



(a) Discrete order picking

(b) Simultaneous zones order picking

(c) Progressive zones order picking

(d) Batch order picking

(e) Batch and zone order picking

(f) Figure explanation

Figure 3.5: Order picking methods
Kay (2015)

3.2 Literature review on VLMs

Meller and Klote (2004) are the first to address analytical models of VLMs. They developed a model to analyse throughput for VLMs by changing multiple variables within the system. These variables include the speed of the AS/R device, the picking time of an item, the height of the VLM and the tray spacing between the trays. Further, they examined how different pod sizes affect the throughput. The authors do not examine how replenishment and put-away would affect the system because the VLMs are designed so that replenishment and put-away should not interfere with order picking. This implies that the replenishment and put-away are executed off-shifts or when the machine is stopped, as stated by Meller and Klote (2004). Dukic et al. (2015) expands the model made by Meller and Klote (2004) to include a dual-bay VLM. Their analysis shows that real distributions of pick time are significant for the system's performance.

Further, the storage assignment policy affects the performance of the VLM. Daria et al. (2015) expanded on the research of VLM by examining how three different storage policies affect the machine's efficiency. They compared random storage, Figure 3.4a, CBS per tray, Figure 3.4b and CBS within trays. CBS per tray lead to the quickest picking time of single-line orders, with the lowest idle time. Nicolas et al. (2018) developed an optimisation model for order batching in multiple scenarios, which is an expansion of Nicolas et al. (2016) model, where they included systems with a single and multiple VLMs. Nicolas et al. (2018) found a correlation between picking time and the VLM with the most visited trays. When the number of orders increases, the probability of compatible orders[1] increases as well, enabling better order batching. However, the probability of finding compatible orders decreased when the number of lines per order increased.

Sgarbossa et al. (2019) included three different sequence strategies of tray retrieval, together with some storage policies, in their throughput model:

- Random retrieval sequence and random storage policy.
- Random retrieval sequence and CBS policy.
- Sequenced retrieval and CBS. The retrievals are sequenced and executed with respect to which SKUs are stored on the trays.

The study concludes with sequenced retrieval and CBS policy being the configuration with the highest throughput for a dual-bay VLM (Sgarbossa et al. 2019). Walitsarangkul and Kittithreerapronchai (2020) explores how different storing zones and retrieval policies of items affect a distribution centre's space utilisation and vehicle utilisation. Vanhauwermeiren et al. (2021) introduces five different VLMs; single bay, dual

[1] orders that are picked from the same tray

bay, double extractor, buffer and independent. The study compares the pick time of these VLMs under different pick and storage policies.

To assess when to use VLMs or not from an economic perspective, Sgarbossa et al. (2017) developed a model to compare a system with VLMs to a system with racks. This is the first paper that addresses how the replenishment activity can affect the system, both from an economic perspective and an operational manner, but only as a further research comment. Calzavara et al. (2019) expands on the economic perspective by including more factors to the model developed by Sgarbossa et al. (2017). Calzavara et al. (2019) developed a model to find when VLM is applicable compared to racks in two case studies. The most important factors to address when choosing between the systems are the cost per square meter, operators' cost, and warehouse saturation level.

Mantel et al. (2007) comes up with a new storage policy, OOS and applies it to a single VLM. The OOS strategy tries to minimise order pick time by storing items that often are picked together close to each other. When using OOS in a VLM, items are put together on the same tray instead of in the same zone or area as in a warehouse using racks (Mantel et al. 2007). One limitation of this study, for the use of VLMs, is that the authors assume the time for changing trays is constant. The model does not take hold of the position of the trays within the VLM.

These findings are summarised in Table 3.3 and Table 3.4. A common feature for all the studies is that they do not look at the replenishment activity, as first stated by Meller and Klote (2004). Further, only three studies have examined a system with multiple VLMs.

Table 3.3: Literature review summary of VLM with type of VLM, number of VLMs and evaluation criteria used in the study.

Year	Authors	Type of VLM	Number of VLM	Evaluation criteriea
2004	Meller and Klote (2004)	Single bay	1; Multiple	Throughput
2007	Mantel et al. (2007)	Single bay	1	Throughput
2015	Daria et al. (2015)	Single bay	1	Throughput
2015	Dukic et al. (2015)	Dual bay	1	Throughput
2016	Nicolas et al. (2016)	Single bay	1	Throughput
2017	Sgarbossa et al. (2017)	Dual bay	1	Economic
2018	Nicolas et al. (2018)	Single bay	1; Multiple	Throughput
2019	Calzavara et al. (2019)	Dual bay	N[2]	Economic
2019	Sgarbossa et al. (2019)	Dual bay	1	Throughput
2020	Walitsarangkul and Kittith-reerapronchai (2020)	Single bay	1	Space and vehicle utilization
2021	Vanhauwermeiren et al. (2021)	Multiple[3]	1	Pick time

[2]Amount depends on multiple factors, including available space, number of SKUs and picks per hour
[3]Single bay; Double bay; Double Extractor; Buffer; Independent VLM

Table 3.4: Literature review summary of VLM with variables used in the study and their limitations.

Authors	**Variables**	**Limitations of study**
Meller and Klote (2004)	Speed; Height; Pod size; Tray spacing; Pick time	Fixed picking time; Excludes replenishment activity
Mantel et al. (2007)	Storage strategy	Fixed tray changing time; Excludes replenishment activity
Daria et al. (2015)	Storage strategy	Picking time is a fixed value; Excludes replenishment activity
Dukic et al. (2015)	Speed; Height	Assumes deterministic or exponentially distributed pick time[4]; Excludes replenishment activity
Nicolas et al. (2016)	Order batching	Excludes replenishment activity
Sgarbossa et al. (2017)	Economic evaluation	Excludes replenishment activity
Nicolas et al. (2018)	Order batching	Excludes replenishment activity
Calzavara et al. (2019)	Economic evaluation	Single order picking; Excludes replenishment activity
Sgarbossa et al. (2019)	Picking time; Height; Storage strategy; Retrieving sequence	Assumes enough inventory to fulfil all orders; Excludes replenishment activity
Walitsarangkul and Kittithreerapronchai (2020)	Number of baskets; Storage policy; Retrieving sequence	Excludes replenishment activity
Vanhauwermeiren et al. (2021)	Types of VLMs; Storage strategy	Picking time is a fixed value[5] Excludes replenishment activity

[4]Authors state that *"it is practically not realistic to expect deterministic pick time per tray (when the number of items to be picked usually varies in practice)."*

[5]Fixed picking time for 5 different product categories

3.3 Summary

Within this chapter, multiple warehouse functions and activities have been described with corresponding policies. Firstly, a warehouse consists of multiple functions and activities, which include replenishment, put-away, storage and order picking. The replenishment activity determines when and how much to replenish, depending on the chosen policy. Four replenishment policies are mentioned; two-bin system, EOQ, ROP and FOI. After the replenishment is decided, the put-away activity has to be executed. The put-away can be executed differently, depending on the chosen practice, strategy and policy. Firstly, two practices are described, the direct and direct secondary put-away, and then three strategies are mentioned; the nondirected, directed and prioritised put-away. Lastly, the put-away can be executed discretely, batched, discretely but within different zones, or batched and between zones. After the put-away is executed, the item is stored. The item can be stored based on various policies. These storage policies include random, dedicated, full turnover, CBS and OOS. Finally, the last activity mentioned is order picking, which can be executed in the same way as put-away; discretely, batched, discretely but within different zones, or batched and zones.

Further, multiple papers are reviewed and analysed to emphasise the research gap of VLMs. These results highlight the missing research on VLMs when multiple are used in sequence. Only three papers mention multiple VLMs; Meller and Klote (2004) investigates the dimensions and speed of the VLMs, Nicolas et al. (2018) investigates the effect of order batching, and Calzavara et al. (2019) carries out economic analysis on when to use VLMs. None of these investigates replenishment, put-away or storage policies. There is currently no research on replenishment and put-away for a system where VLMs are used, neither a single VLM nor multiple.

Chapter 4

Logistikksenter Helse Midt-Norge

Most of the content in this chapter is directly cited from an unpublished report written as the final assessment in the course *TPK4530* given at the NTNU. The author participated in this course during the autumn of 2022.

As outlined in chapter 1, HMN has made substantial investments in the establishment of LS HMN, a state-of-the-art warehouse facility. In contrast to the previous decentralized model where hospital warehouses were dispersed throughout the region, the LS HMN warehouse represents a centralized approach, catering to the needs of all regional hospitals. This newly developed facility is designed to serve the region in the foreseeable future, equipped with advanced warehouse technology and modernized ICT systems.

LS HMN primarily functions as a storage and distribution centre for non-pharmaceutical hospital supplies. With a wide range of 4500 SKUs, there is a need for a spacious reserve storage area to ensure the continuous availability of given hospital supplies. Additionally, there is a dedicated forward picking area with multiple VLMs, enabling immediate access to all SKUs for efficient picking operations. To achieve the objective of supplying the region with hospital goods, LS HMN is equipped with the following components and features:

- Multiple pallet racks used for reserve storage. The first and second floor of each rack is used as a forward picking area for larger items and for the items that do not fit within the VLMs.
- A storage area for flammable items.
- An emergency storage.
- 23 VLMs used as the main forward picking area. These are split into three zones depending on the items' characteristics; sterile, non-sterile and picked totes.

4.1 Warehouse description

4.1.1 Material flow

The warehouse is split into seven main areas, from inbound storage to outbound storage. The material flow follows the typical warehouse material flow, as introduced in chapter 3. The layout and material flow are depicted in Figure 4.1 and have the following seven steps:

1. Items are received at the inbound gates and put into the inbound storage area. A check of the inbound items is carried out to ensure the right SKU, the right amount and the right quality.
2. After the check, the items are sent into storage depending on their characteristics and needs. Items that do not require any specific environment while in storage are stored in the pallet racks. Some items are sent to an emergency storage area. These SKUs are rotated in order to ensure that there are no expired or soon-expiring items. Lastly, the items that are highly flammable, and hence in need of a specialized area, are sent to a specialized area for this.
3. When the VLMs need to be replenished, the pallet of given SKU is transferred to the de-palletizing area. Here the items are separated from one pallet into individual cases.
4. The individual cases then go to different picking and storage locations, depending on their characteristics. Items that do not have any specific requirements go into the normal VLMs, *4A*, and the sterile items are transferred into the sterile zone, *4B*.
5. Order picking is executed in the three main areas and transported to the consolidation area; *5A*, *5B* and *5C*. The different orders are put together and placed into their corresponding transportation carts in the consolidation area.
6. The transportation carts are put into outbound storage and ready to be picked up.
7. The transportation carts are loaded into trucks and delivered to their destination.

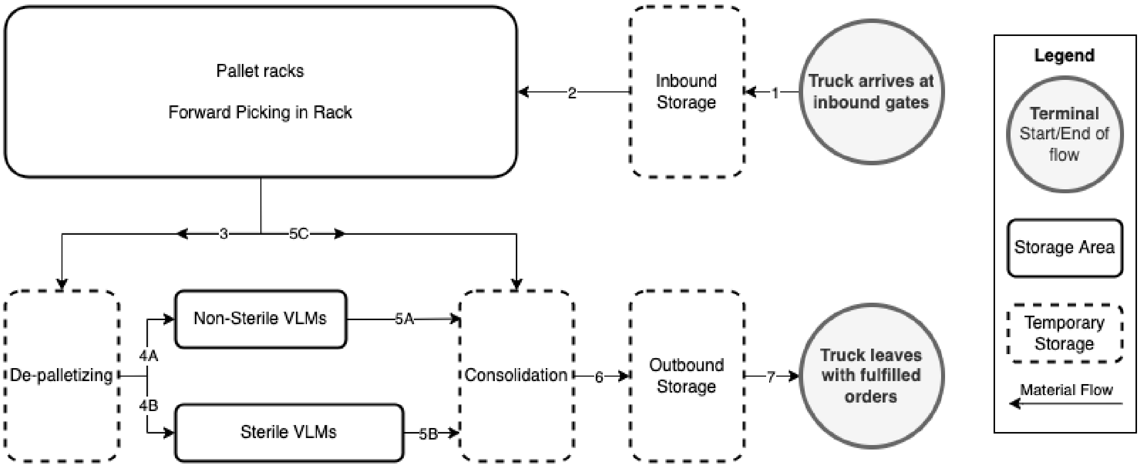


Figure 4.1: Conseptual model of the layout and material flow of LS HMN

4.1.2 VLMs

The VLMs are split into three groups; sterile, non-sterile and picked totes. There are 12 VLMs operating in sequence within the sterile zone. When the picking of an order begins, the VLM with the first item (item N) begins to extract the tray with item N. Simultaneously, the VLMs with item N+1 and N+2 begins to extract the trays with item N+1 and N+2. This is the *active VLMs parameter* and it is applied to reduce waiting for the operator at the VLM. Usually, this parameter is set to three. When the VLM receives multiple instructions about which tray to fetch, the extraction process is queued. The sequencing of retrieval of trays follows a First in first out (FIFO) rule for the queue.

The active VLMs parameter is only applied to order picking. When a put-away is executed, only the VLM which should be replenished is receiving instructions. As soon as the put-away is executed, the next VLM receives instructions. When the trays are put back into the racks, the height of the boxes and tray is scanned and the tray is inserted at the level which optimises storage volume usage.

4.1.3 Warehouse policy for each warehouse function and activity

Replenishment

A FOI replenishment policy is generally used in all the warehouse zones and the stock count is carried out daily by the WMS. If the storage level of a SKU is below a predefined level, a replenishment order is generated. Within the racks, the WMS suggests a location for the given SKU, and the operator confirms the correct location once the put-away is executed. In the sterile and non-sterile zones, multiple items can be loaded onto a picking/put-away cart, and the operator performs the put-away in the correct

VLM and slot on the given tray. The WMS decides exactly where the item should be stored.

Put-away

The put-away activity is carried out as soon as the replenishment order is available, and it is executed with a zone policy in parallel with order picking. When the items of a single SKU in the replenishment order are available, the operator loads these items onto the picking/put-away cart. When the operator is at the correct VLM and the correct tray is displayed, the operator executes the put-away. The correct slot is shown by a pick-to-light system. For the VLM handling picked totes, the replenishment is carried out through the back of the VLM, contrary from the front. LS HMN uses a directed secondary put-away. The items are stored in a reserve storage area before replenishing them into the VLMs, known as a direct secondary put-away. Further, the WMS decides where to put the items, known as a directed put-away.

Storage

Products are stored in zones, determined by their characteristics, either sterile, non-sterile or picked totes. Within the zones, the items are spread across multiple VLMs to ensure better availability of the items. Products are stored with a random storage policy within the VLMs. When the WMS is finding a slot for an item, it starts in VLM 1, tray 1 and slot 1. It goes through all the slots in tray 1, and if there are no available slots, the WMS checks tray 1 in VLM 2 and so on. If no slots are available in tray 1, then tray 2 is checked. The item is assigned to the first available slot that the WMS finds.

Order picking and releasing

LS HMN applies a wave-picking procedure with zone picking. Wave-picking is when multiple orders are released based on a time cycle (de Koster et al. 2007). When an order arrives, it is grouped with orders from the same customer with the same due date and consolidated into a delivery order. The delivery order is then split into multiple picking orders for the different zones, i.e. racks, non-sterile, sterile and picked totes. Picking lists are created within each zone, where one list corresponds to one customer and one operator can only have one list at a time. When the picking lists for the given order are fulfilled, the order is consolidated with items from each zone and placed within a transportation cart. The transportation carts are labelled with destination, and multiple customer orders can be consolidated within one transportation cart as long as they have the same destination. The whole order processing process is illustrated in Figure 4.2.

The main picking policy at the racks is discrete order picking, with the possibility of batch picking depending on the volume and size of orders. A zone-picking policy is induced within the VLM zones where the different zones are the sterile and non-sterile areas. In these zones, items are picked onto picking/put-away carts where one operator operates one cart, and one order is linked to this cart. LS HMN have induced a FIFO rule that control which individual item of a given SKU should be picked first. The logic ensures that the items that have been in storage the longest are picked first. On perishable products, a First expire first out (FEFO) rule is used in parallel with the FIFO. FEFO determines that the products with the least days left before they are termed expired are picked first.

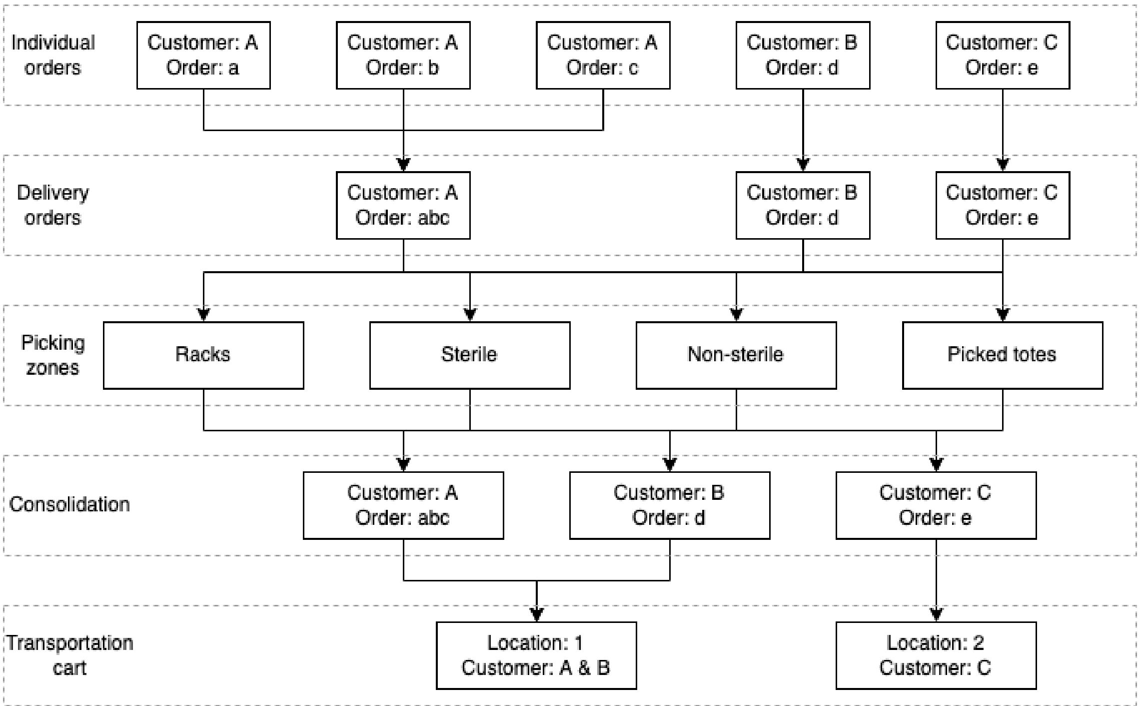


Figure 4.2: The order flow and zone classifications used at LS HMN

Lastly, this FIFO logic is also applied to the sequencing of tasks for the operators. If there is a queue of available orders before a replenishment order is released, the orders that arrived before the replenishment order will be picked before the replenishment order is executed. This also applies to the replenishment, if multiple replenishment orders are queued when an order is released, the replenishment orders will be executed first. Also, this logic applies to the VLMs. When the VLM is given instructions to fetch trays, the sequencing of retrieval will be based on FIFO.

4.2 Applicable warehouse policies

4.2.1 Replenishment

Table 3.4 and Table 3.3 show that not a single study has looked at the replenishment activity. All the previous studies assume that there is always enough stock to fulfil the orders or that replenishment is done off-shift, as stated by Meller and Klote (2004). However, when the replenishment has to be executed in parallel, it is crucial to find different replenishment policies. There are multiple different replenishment policies, as introduced in Section 3.1.1. These can all be implemented into a system with multiple VLMs:

- The **two-bin system** can be introduced to the individual trays within a VLM or between the total stock of each VLM. When a SKU is emptied from a tray or VLM, a replenishment order is placed with a specific quantity and another tray or VLM is used until it is emptied. The current storage could be verified and updated by the operator, but ultimately the WMS should keep track of this information.
- The **ROP** policy can be introduced to individual VLMs stock or to the stock of all the VLMs combined, further referred to as the total stock.
- The **FOI** policy can be introduced to individual VLMs or the total stock.

The level of where the replenishment policy is placed can affect the system in multiple ways. The level is where the replenishment policy is placed, either at individual VLMs or at the total stock. When the policy is set to the total stock of the system, the order size can increase, and the frequency will then decrease, compared to individual VLM storage. Instead of having one order for each VLM, it will be one large order for the whole system. When the policy is set at the individual VLM, the ROP might be reached more often with smaller order sizes. The frequency for FOI will be consistent, but the amount of replenishment orders might decrease.

Further, if the policy is applied to the total stock, the availability of items can go down because multiple trays and VLMs can be emptied before a replenishment order is placed. If combined with dedicated storage, trays and VLMs can be emptied for longer, compared to CBS or random storage. If CBS or random storage is used, the SKUs stored within the VLMs are not predefined and the slots can be replenished whenever a replenishment order is placed, no matter which SKU is replenished.

Currently, LS HMN applies a FOI policy for its VLMs, with the total stock as the replenishment level. Table 4.1 summarizes the current replenishment policy at LS HMN and the different available replenishment policies.

Table 4.1: Replenishment policy at LS HMN and other applicable policies

LS HMN	Other applicable policies	
FOI	Two-bin system	Individual tray Individual VLM
	ROP	Individual VLM Total stock
	FOI	Individual VLM Total stock

4.2.2 Put-Away

There are multiple ways of organizing put-away, which are deducted from the different order picking methods mentioned in Section 3.1.4 (Frazelle 2016; Kay 2015). For LS HMN, the activity should be organized in the same zones as order picking, e.g. sterile and non-sterile, to reduce the need of transferring operators between the sterile and non-sterile zone. LS HMN uses a zone configuration for their put-away where one operator can execute one replenishment order at a time. A summary of this small discussion is seen in Table 4.2.

Table 4.2: Put-away policy at LS HMN and other applicable policies

LS HMN	Other applicable policies
Zone	Zone Zone-Batch

4.2.3 Storage policies

A storage policy decides where and how to store the items, and multiple policies are applicable to VLMs. Random storage will optimize storage utilization by allowing all products to be stored wherever available, while dedicated storage will use predefined locations for all products. Dedicated storage occupies more slots because each slot is dedicated to a specific SKU, and the slot cannot be used by another SKU even if the slot is empty. Further, a full turnover storage policy could be used. Here the location of the SKUs can change based on the fluctuations in demand (de Koster et al. 2007). CBS, the combination of random, dedicated and full turnover storage is another alternative. When CBS is used, the products are grouped based on their turnover rate, each group is given a dedicated area and the items are stored randomly within its given group´s area. A fifth option is the family grouping policy, OOS, where items are stored based on their relationship.

These storage policies are all applicable to warehouses, no matter their storage technology. Applied to a system with a single VLM, these policies decide which trays should store which products and it is shown that the policy affects the throughput (Daria et al.

2015; Mantel et al. 2007; Sgarbossa et al. 2019). However, these possible configurations are not yet researched in a system with multiple VLMs.

Random storage will utilize the storage capacity of the system, but it might reduce the extraction time of trays compared to other storage policies (Daria et al. 2015). For example, retrieving a set of high-demand products may take longer than other storage policies because the products can be stored far away from the bay. With dedicated storage, products or product groups can be assigned to different VLMs where only these products can be stored. Multiple VLMs can be configured to have identical dedicated storage locations. This dedicated storage configuration might reduce availability and increase the picking time of the products stored in the same VLM because the operator has to wait for the VLM to retrieve the correct tray. An advantage of this is that the operator knows which VLMs have which products, and it will reduce the travelling time needed. However, it can lead to low storage utilization because empty locations are assigned to products not in stock, as discussed by de Koster et al. (2007) and Kay (2015). Also, at LS HMN, the operators have a terminal which notifies the operator which VLM the following order line is in.

The CBS policy can help reduce the retrieval time because the trays with high turnover rates are close to the picking bay Daria et al. (2015). This policy can be configured in three main ways:

1. **Categorize the VLMs**. A few VLMs with high-demand products, while the rest have low-demand products. The high demand VLMs will be visited often, and the picker have to wait if two consecutive items in the order are placed in the same VLM but on different trays. With this policy, the pickers know in which VLMs the different SKUs are stored and the searching time can be reduced.

2. **Categorize the trays within each VLM**. All VLMs share the same storage configurations, where high-demand products are stored close to the picking bay and low-demand products are stored further away (Daria et al. 2015). The picker does not know where the products are stored, but the retrieval of high-demand products will be quicker.

3. **Categorize the slots within each tray**. This configuration was introduced by Daria et al. (2015), and they found out that the configuration led to a reduced throughput compared to random or the CBS configuration mentioned previously.

Further, the active VLMs parameter can reduce the waiting time for the operator since the tray might already be displayed when the operator arrives at the VLM. The effect of this might be more apparent when items are stored far away from the picking bay. For random storage, this can be any item and for CBS, this would be the items with low demand. The effect might not be as apparent if OOS is applied and if orders can be completed by picking from one tray.

LS HMN uses a storage system with three zones based on the requirements and characteristics of the products; sterile, non-sterile and picked totes. Sterile products must be stored in a sterile zone due to their characteristics, while picked totes need their own zone due to the different material handling methods. Further, within the zones, a random storage policy is used to evenly spread the products between the VLMs. Table 4.3 shows LS HMN current storage policy and other applicable policies. How these affect the order picking time and throughput of a single VLM is already researched, see Section 3.2, but how they affect a system with multiple VLMs are yet to be researched.

Table 4.3: Storage policies at LS HMN and other applicable policies

	LS HMN	**Other applicable policies**	
Zones	Sterile, Non-Sterile and Picked Totes	-	
VLMs	Random	Random Dedicated OOS	
		CBS	Within each VLM Between VLMs

4.2.4 Order picking

Order picking is the most cost- and time-consuming activity of a warehouse, and therefore the picking time should be minimised (de Koster et al. 2007). There are multiple methods of organising order picking, which determines how many pickers per order and how many orders per picker. Discrete is the simplest, where one picker picks one order at a time. However, the total pick time for multiple orders will increase because the picker will travel to the same locations multiple times. To counteract this, a batch-picking method could be applied. This enables the same picker to pick multiple orders simultaneously in batches. However, both discrete and batch do not consider the different product characteristics or zones, such as LS HMN have. To move between the sterile and non-sterile zones, the picker has to undergo an extensive process of becoming sterile. Zone picking could be applied to remove this process. There will be multiple pickers per order, but each picker only picks a part of the order. Within the zones, discrete picking is used. To further reduce travelling time, zone picking can be combined with batch picking, where a picker confined within its zone can pick multiple orders simultaneously.

LS HMN applies a zone-picking policy to reduce the need for order pickers to travel between the sterile and the non-sterile zones. The process will not be eliminated but executed at a minimal rate. For example, the process would still be executed at the start and end of shifts or because of breaks. The incoming orders are split into each zone, and each picker then picks one order based on the generated picking lists. Discrete picking will not work at LS HMN because the items in the order are spread across the different zones, especially the sterile and non-sterile zone. An order containing sterile and non-sterile items must be picked in their corresponding zones. However, this can be done either discretely or batched, see Table 4.4.

Table 4.4: Order picking policy at LS HMN and other applicable policies

LS HMN	**Other applicable policies**
Zone	Zone Zone-batch

4.3 Summary

In this chapter, a detailed analysis of LS HMN has been carried out. The material flow and a conceptual model of the warehouse are shown with the corresponding functions and activities within each part of the warehouse. Also, the warehouse policies applied at LS HMN are listed. After the description of LS HMN follows a small discussion regarding what the applicable policies are for a system with multiple VLMs. For the decision of when to replenish the storage, ROP and FOI could be used with two different configurations, either a total stock for the whole system or an individual stock per VLM. When it comes to the two-bin system, one could use either individual trays or VLMs as the bins. The put-away could be executed in two main ways, either discretely by zone or by zone with a batch method. When it comes to storage policies, there is only one configuration for LS HMN with its zones, and that is sterile, non-sterile and picked totes. This is due to the product's characteristics. However, for the storage within the VLMs there are multiple policies. Random, dedicated or OOS could be used by itself, or they can be combined into the CBS policy. CBS could be applied either between the VLMs, or within each VLM. When it comes to order picking, there are only two ways for LS HMN, either zone or zone and batch. These concluding remarks are shown in Table 4.5, which also includes LS HMN current policy for the different warehouse functions.

Table 4.5: Current warehouse policies at LS HMN and other applicable policies

Warehouse functions and activities	**LS HMN**		**Other applicable policies**	
Replenishment		FOI	Two-bin system	Individual tray Each VLM
			ROP FOI	Total stock Individual VLM
Put-away		Zone	Zone Zone-Batch	
Storage policy	Zones	Sterile, Non-sterile and Picked totes	-	
	VLMs	Random	Random Dedicated OOS	
			CBS	Within each VLM Between VLMs
Order picking		Zone	Zone Zone-batch	

Chapter 5

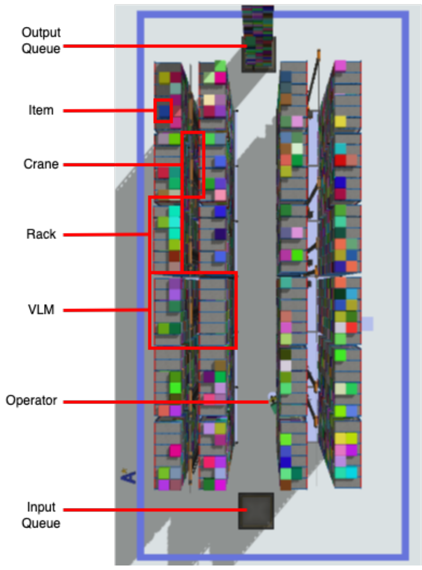
The Simulation Model

The DES model built in FlexSim® is a replica of LS HMNs sterile zone. It includes the objects and logic needed to simulate how changes in logical parameters the system. All the physical parts of the system are included in the 3D model. The 3D model is based on drawings, pictures and measurements of the real system, making the visual model as realistic as possible.

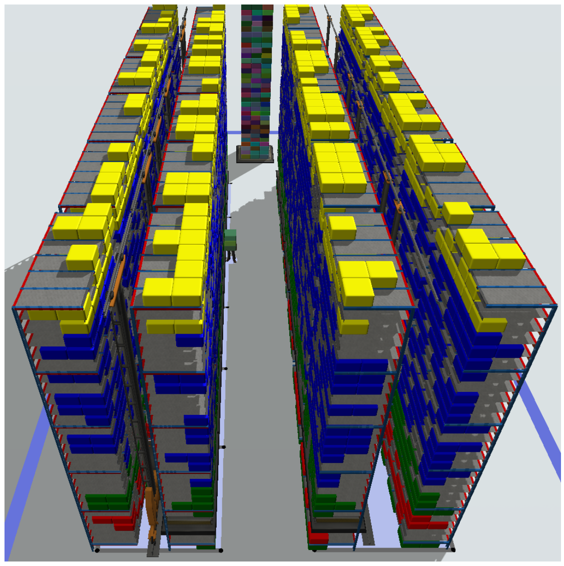
Further, the model uses data extracted from LS HMNs WMS to control the initial inventory and order generation. All other logic is programmed based on the information given from LS HMN and other sources, such as MHI (2023b) and Sgarbossa et al. (2019). The model includes all the central components of the sterile zone needed to create a DES where the effect of changes in logical parameters can be analysed. The logic of the model is designed to be parsimonious, which implies that it is kept as simple as possible while still fulfilling its intended purpose (Sargent 2020). The 3D model includes the following:

- 12 VLMs used for storage and order picking.
- Multiple order pickers.
- An input location where items are spawned when they are replenished.
- A output location where items are stored after they are picked.

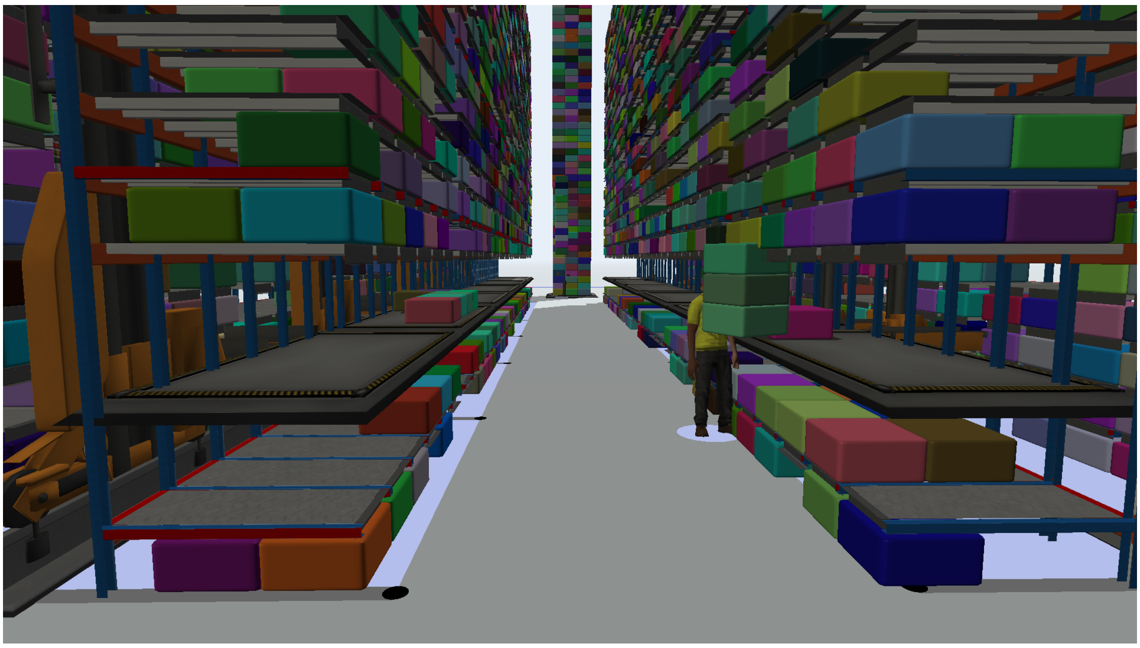
The VLMs are modelled as two racks with an AS/R device between, which acts like a crane. The picking bay is modelled as a queue where items are stored temporally. The input and output locations are modelled as queues as well. The number of operators is adjusted according to the current scenario, but it is usually only two within the model.



(a) Model view from the top with explanations



(b) Model view from the front with a CBS policy



(c) Close up frot view with a random storage policy

Figure 5.1: Multiple model views with graphical explanation and different storage policies.

5.1 Logic of the model

Activities and resources within the Process Flow control the model's logic. The activities and resources are also connected to the objects within the 3D model and give instructions to the cranes and operators, known as the task executors. The Process Flow does all the calculations and decides when and where the task executors should move and which objects or items they should interact with. Once the task executor receives a task, e.g. walk to a VLM or retrieve a specific item, the 3D model controls the movement of the task executor. The speed at which operations are executed, e.g. how long it takes for the operator to move from one VLM to another VLM, is decided by the 3D model logic and not the Process Flow.

The logic of the model is split into four main parts:

1. Initial Stock - Creates the initial stock and decides where the items should be stored
2. Order Releasing and Order Picking - Finds items for each order and executes the order picking
3. Item Generation - Spawns items for replenishment and controls the replenishment activity.
4. VLM - Controls the VLMs

5.1.1 Initial stock

When the simulation starts, a token is generated at the "At Simulation Start" activity, shown in Figure 5.2. Then the token goes to the "Generate Items" activity, which loops through a table of the initial stock. First, the activity creates an item with a SKU and quantity, predetermined by the initial stock table, which is given from LS HMN. Then it finds a slot for the item in one of the racks, either randomly or based on the CBS policy, and changes the item's colour to fit the policy. See Figure 5.1b and Figure 5.1c for examples of this. Lastly, information regarding the item's location is stored on the item, the total stock count is updated, and the item is teleported to its correct slot.

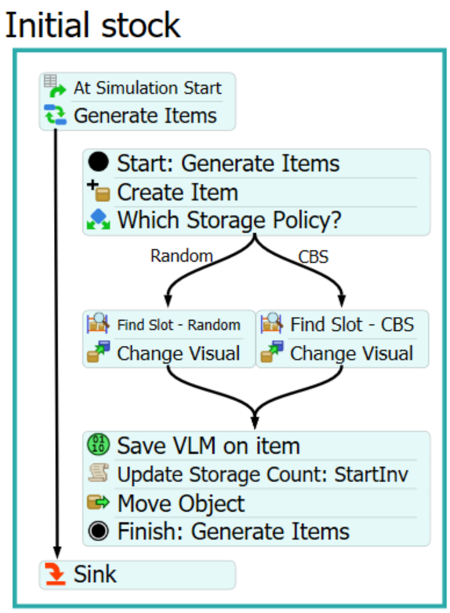


Figure 5.2: Initial stock generation in Process Flow

5.1.2 Order releasing and picking

After the initial stock has been spawned, the order-releasing process begins. Figure 5.4 shows a simplified flow of the order-releasing logic, and the complete process flow is shown in Appendix A. Further, Figure 5.3 shows the legend of the flow charts in the rest of this chapter.

First, an order is created and released at the given order's start time. When the order is released, the items within the order are located and assigned to the order. This process is the *Fill Out SKU Line Items* in Appendix A. After the item is assigned, the stock level is updated, and the ROP policy is checked if it is applied. If ROP policy is applied and the stock level has reached the ROP, a replenishment order is spawned. The replenishment process is shown in Section 5.1.3. This cycle repeats until the order requirements have been fulfilled. After the order requirements are fulfilled, the order picking process starts. Then the following order is created and waits to be released. This process repeats until all the orders are released.

A simplified flow chart of the order picking process is shown in Figure 5.5, and the complete process flow is shown in the *Do Batch Picking* and *Pick Items* blocks in Appendix A. When the order picking process start, the orders could be batched or picked discretely. First, an operator is acquired, if available. The process waits here until an operator is available. Then the VLM with the first item is given instructions to fetch the tray with the item. Simultaneously, the VLMs with item N+1, N+2 up to the active VLMs parameter, is given instructions to fetch the tray with item N+1, N+2 up to N+Parameter, as introduced in Section 4.1.2. The rest of the VLM process is shown in Section 5.1.4.

The operator then travels to the VLM with item N and waits until the tray with the item has been displayed in the picking bay. The item is then picked, and if the order is not fulfilled, N is increased by one, and the cycle repeats until the order is fulfilled. Once the order picking is finished, the operator travels to the drop-off location to drop off the items. Statistics regarding the individual picks and the operator is recorded, and the operator is released.

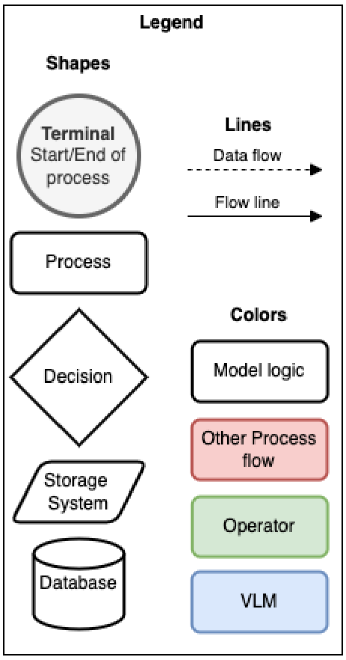


Figure 5.3: Legend for the simplified process flows of the model logic

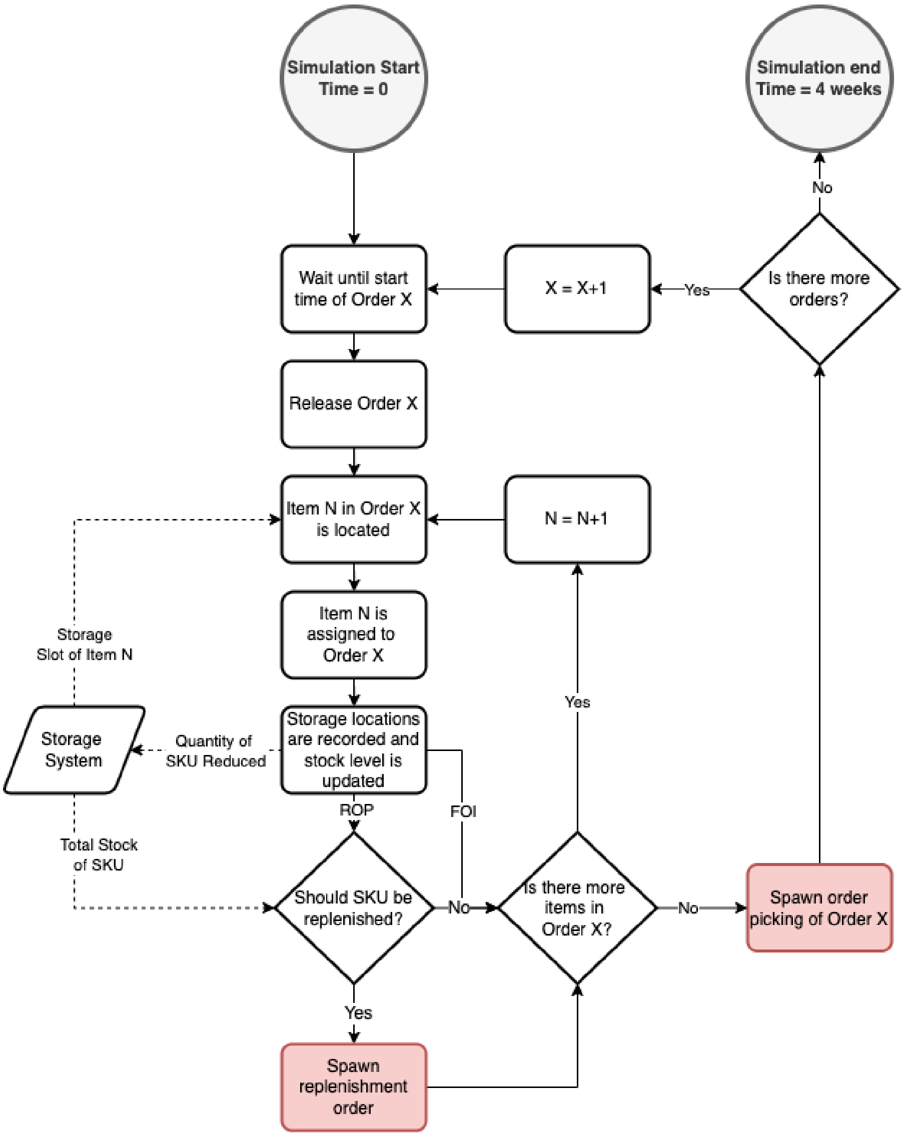


Figure 5.4: Order Releasing process flow

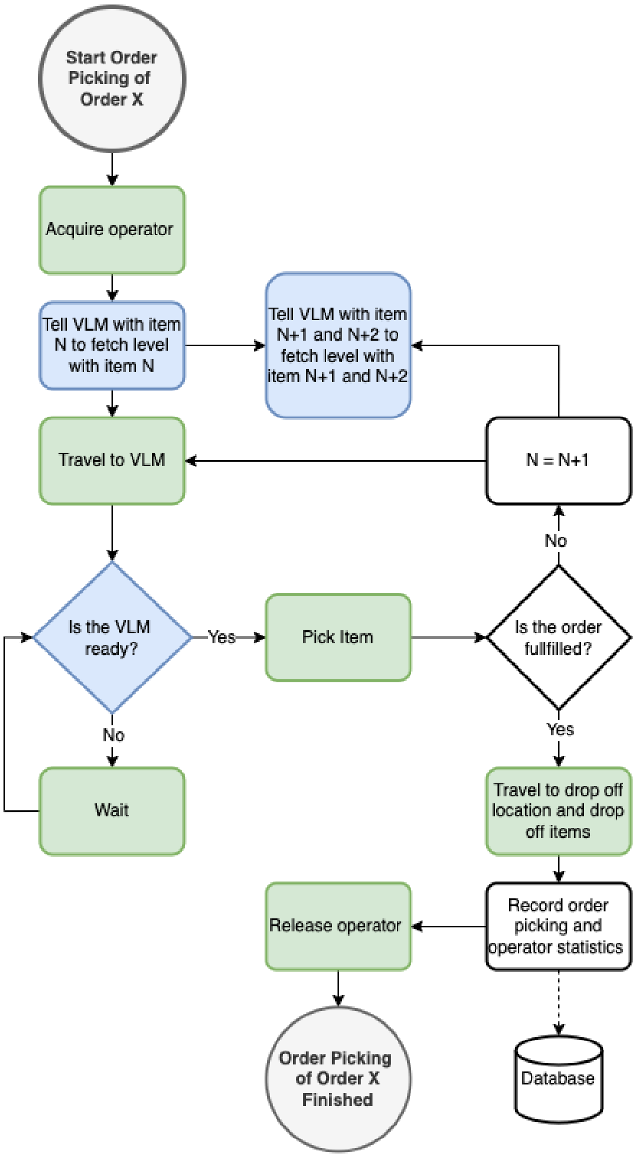


Figure 5.5: Order Picking process flow

5.1.3 Replenishment

The replenishment activity is initialized as soon as a replenishment order is created. A simplified process flow is seen in Figure 5.7 and the full process flow is seen in Appendix A. A replenishment order can be created in two ways, either from a ROP or based on a FOI policy. When the replenishment order is spawned, the SKU and given quantity of the replenishment, decided by ROP or FOI, is spawned at the input location and the items are immediately given a storage slot based on the current storage policy.

After the items are given a storage location, an operator is acquired, if available. The process waits here until an operator is available. The operator then travels to the input location and picks up either a discrete item or a batch of multiple items. Then the VLM with the tray that should be replenished is given instructions to fetch this tray. The VLM process is described in Section 5.1.4. When the VLM is ready, the operator puts the item in the correct tray, and this cycle repeats until all the picked-up items are put away. Then the statistics of the replenishment are recorded, the stock is updated, and the operator is released.

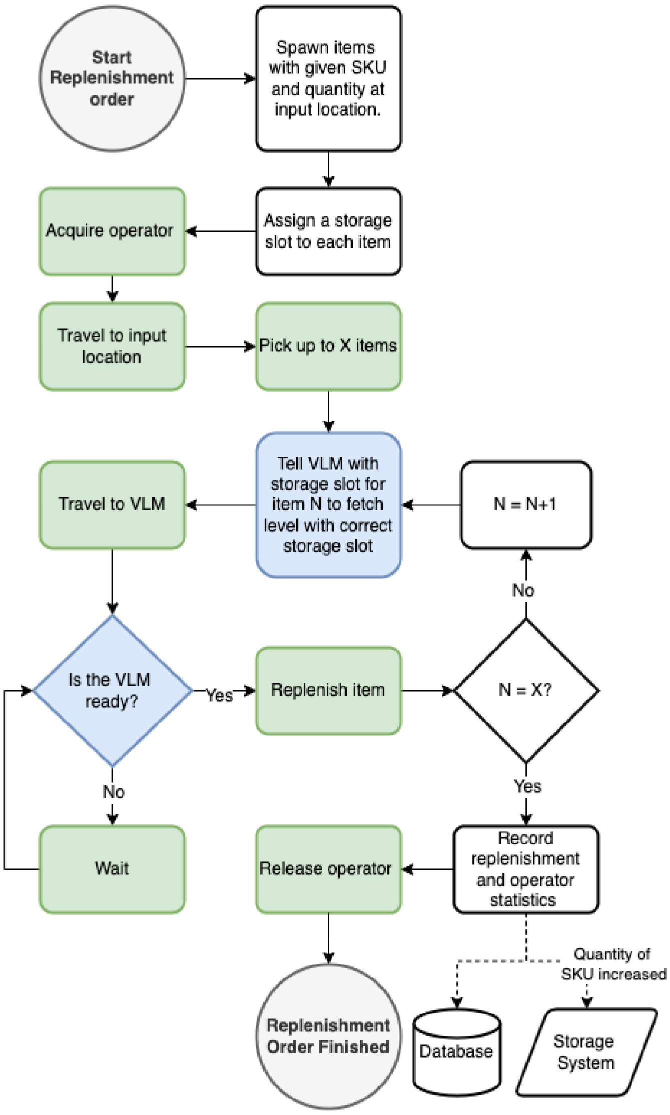


Figure 5.6: Replenishment process flow

5.1.4 VLM

The VLM process starts as soon as an item should be picked or replenished, as mentioned earlier. A simplified process flow is seen in Figure 5.7 and the full process flow is seen in Appendix A. First, the crane moves to the correct level in the rack. Then, a single item is loaded onto the crane, which moves to the picking bay and unloads the item. Immediately after the single item is unloaded at the picking bay, the remaining items in the level are teleported to the picking bay. This picking up one item and teleportation the rest of the items process replicates a real VLM which picks the whole tray. Once the items are teleported to the bay, the operator can either pick or replenish an item. When the operator is finished, a single item is loaded onto the crane. The item is then unloaded on the same level as it started, and the rest of the items at the picking bay are immediately teleported. The VLM is finished, and the next task can be received.

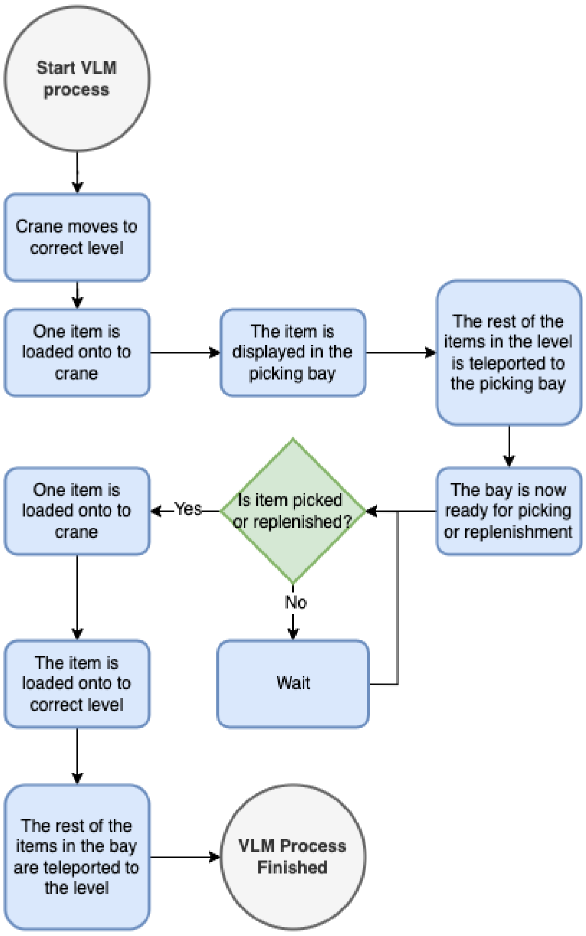


Figure 5.7: VLM process flow

5.2 Imported and exported data

5.2.1 Imported data

The models' initial inventory, order generation, and info about the SKUs are based on reports extracted from LS HMNs WMS. The initial inventory consisted of all the occupied storage locations at LS HMN, with its corresponding SKU and quantity. The storage locations were removed when the data were imported into the model, and only SKUs, and quantities were imported. The storage locations were removed because the model is supposed to find locations for the items based on either a random storage policy or CBS policy.

Four weeks of orders are imported and simulated. The orders consist of an order-id which includes all the SKUs with corresponding quantities, and the order releasing time. The order releasing time is the time when the first item in the order is picked in the real system. The orders are imported this way to ensure that the orders within the model are identical to those picked in the real world. In addition, some information regarding the items was also imported into the model. This additional information includes the maximum quantity per storage location, start quantity of each SKU, a ROP, how much to replenish and the CBS classification of the item.

The classification of items was calculated based on the maximum amount of needed slots per SKU during the simulated period. The *A* items, which are located closest to the picking bay and coloured red in Figure 5.1b, are the items which need the highest amount of slots and not the quantity. The quantity of some items might be in the thousands but only require one slot throughout the simulated period, while other items might only be a couple of hundred but requires multiple slots during the simulated period. Further, the *D* items, located on the top of the VLMs and coloured yellow in Figure 5.1b, are the items which are not requested during the simulated period but are included in the initial inventory. *B* items are green and *C* items are blue.

5.2.2 Exported data

Multiple data sets were exported in order to be able to analyse and compare the different scenarios. The system's stock, in quantities and not occupied slots, is recorded every hour for each VLM and total stock. This data is used for verification and validation, see Section 5.5. Further, the stay time for an order or replenishment order within the model is recorded. A timer starts once the order is available to be picked or replenished. The timer stops when the order is fulfilled and dropped off, or when the replenishment order and put-away is finished. This is referred to as the total stay time for an order. This recording is used to see how long an order is within the system, from available until finished. This timer will increase if there is a queue of multiple orders or replenishment orders, or if all the operators are unavailable.

Every time an item is picked or replenished, the time it takes to complete this task is recorded. This recording includes multiple time measures:

- First, the time for the operator to walk to a specific location is measured. Usually, this is the time it takes between two VLMs.
- Then, the waiting time for the operator is recorded. This is when the operator has to wait for the VLM to fetch the correct tray after the operator arrives at the VLM.
- The time it takes for the operator to execute the activity is recorded.
- Order- or replenishment-id, which operator, and which VLM is also recorded.

These time measures determine how long it takes to process a task, whether a single order line or the whole order, from received and available until completed. Further, the deleted order lines are also recorded with all the information regarding the SKU, quantity, order-ID, order line number and recorded time stamps.

5.3 Model parameters

Multiple parameters are used within the model. The parameters are split into three main groups:

- The fixed model parameters for VLM. These are the same values as those used in the real system. The parameters are shown in Table 5.1 and are used for creating the VLMs.
- The fixed model parameters for the operators, see Table 5.2. These are used to model the operators. The loading and unloading times are based on observations of the actual system.
- The variable model parameters are shown in Table 5.3. These are related to the different scenarios and what values can change between them. ROP is set to this to trigger the replenishment at a higher frequency than if it was set to a lower value. The FOI is the same as the real system.

Table 5.1: Fixed model parameters - VLM

Parameter	**Value**	**Unit Measure**
VLMs	12	
Height of VLM	10.8	m
Width of VLM	3.72	m
Picking bay height	1	m
Width between VLM rows	3	m
Total length of VLM row	10	m
Trays per VLM	51[1]	
Slots per tray	12[2]	
Total storage slots of system	7200	
Items per slot	1	
Size of slot	0.93	m^2
Height of slot	0.4	m
Initial crane height	1	m
Vertical speed of crane *empty*	0.76	m/s
Vertical speed of crane *loaded*	0.42	m/s
Acceleration and deceleration of crane	1	m/s^2
Horizontal crane speed	0.29	m/s
Empty tray receiving time	30	s

Table 5.2: Fixed model parameters - Operator

Parameter	**Value**	**Unit Measure**
Operator speed	1.5	m/s
Operator load time *picking* per item[3]	$\mu = 21, \sigma = 2$	s
Operator unload time *picking* per order[4]	30	s
Operator load time *replenishment* per order[4]	30	s
Operator unload time *replenishment* per item[3]	$\mu = 21, \sigma = 2$	s

[1] 24 in the rack with the picking bay and 27 in the other rack
[2] Six in width and two in-depth (6x2)

Table 5.3: Variable model parameters

Parameter	Value
ROP	Half of the quantity of the initial inventory
FOI	Once a day
Order picking batching amount	1 or 5
Replenishment items batching amount	1 or 10
Wave releasing	Once every hour
Active VLMs	Between 1 and 5
Amount of operators	Between 1 and 5

When the FOI occurs, the model checks the total stock of each SKU, either for the total stock or for each VLM, depending on the replenishment level. A replenishment order is spawned if the initial and current stock difference exceeds the slot quantity for the given SKU. The amount of slots to replenish is determined by Equation 5.1. The *floor* function returns the largest integer value not larger than the number, and it is included to ensure that only an integer amount of slots can be replenished (FlexSim® 2023b). The *+1* is included to ensure that the stock after replenishment will be as close to the initial stock as possible. A *round-up* function will give the same results.

$$\text{Slots to Replenish} = \textit{floor}\left(\frac{\text{Quantity to Replenish}}{\text{Slot Quantity}}\right) + 1 \tag{5.1}$$

[3]Normal distributed.
[4]Constant value no matter order size.

5.4 Model limitations and simplifications

Although the model strives to be as realistic as possible, some simplifications and assumptions are included in the model logic. The first simplification is that the boxes or items within the model all have the same size and take up the same volume, regardless of the quantity stored or which product it is. In the real system, the size of the boxes depends on the products. The size of the items within the model is set to a standard to reduce the data needed for the simulation and to simplify the storage constraints in each tray. A tray can store up to 12 items, so each item is 1/12 of the size of a tray. Further, within the real system, the trays within each VLM is stored and reorganized to maximize volume utilization. The height between trays is determined by the height of the boxes and not a fixed height. This logic is not included in the model. In the model, each tray has its specific level in the rack. When the tray is inserted into the rack, the tray is always given the same level.

In the simulation, only one item can be picked from a tray each time the tray is displayed. If multiple items in one order are stored on the same tray, the tray will be stored and displayed between each pick. The trays will also be stored and displayed between picks if the same tray should be picked from multiple orders simultaneously. In the real system, the tray is left at the picking bay until all the items are picked. In the simulation, because of this logic, the operators might have to wait longer and more often on the VLM compared to the real system.

Sometimes the items within an order do not exist within the model because they have not been replenished yet. If this happens, the order releasing flow, Figure 5.4, will try to locate the item for some time. If the item is not located within a predefined period, set to 1 hour, the order line will be removed from the order. Removing order lines might happen to orders with only one line, and then the order is deleted. Overall this does not affect the system performance due to the scarcity of this event. Less than 1% of all order lines are deleted because of this logic.

When the operators pick or replenish an item at LS HMN, they scan multiple barcodes and put the item on the picking cart or the tray. However, in the simulation, they stand still and wait approximately the time it takes for the scanning and picking or put-away. When the timer ends, a new item is spawned in the operator's arms with the same data as stored on the item in the VLM, or the item is placed in the correct slot in the tray if it is a replenishment. If the quantity of the picked item is zero, the item within the VLM is deleted. Otherwise, the quantity is reduced by the order quantity. This logic is implemented to ensure that the same item in the model can be picked multiple times due to the different quantities stored within each item. The quantity of the item is reduced when the item is linked to the order, see Figure 5.4, and it is deleted when the operator picks the item that reduced the quantity to zero.

5.5 Verification and validation

Multiple steps have been conducted to verify and validate the model. The verification, ensuring that the model operates as it should, has been conducted continuously during the development. For instance, at the beginning of the development, the operators climbed an invisible staircase to pick the items in the top level of the VLM instead of picking them from the bay. Another problem halfway through the development was that the cranes suddenly unloaded the items a few meters from the bay. These errors were fixed as soon as they occurred. Multiple visual checks of the Process Flow and 3D model have also been conducted during the development to fix bugs. Towards the end of the development, when the replenishment policy was introduced, the stock count was recorded to ensure that the model did not delete or spawn more items than it was supposed to do.

Figure 5.8 shows the total stock of the model compared to the real stock. The real stock is calculated based on the initial stock, the incoming replenishment orders and the outgoing customer orders, all given from LS HMN. The flat area in the simulation stock is a week where no orders were picked or items replenished. The model does not use real data for replenishment, it is supposed to decide when to replenish. When no orders were picked, LS HMN decided to replenish multiple items, increasing the stock levels. These replenishments are not taken hold of in the model, and therefore an adjusted stock is calculated. In the adjusted stock, the highest peak of replenishment during the period where no orders are picked is removed. When the adjusted stock and the stock in the simulation are compared, it is noticeable that the stock level follows the same trends and is generally at the same level.

Further, the order lines picked per hour, see Figure 5.9, is roughly the same between the model and the real system. The real system has higher peaks in the beginning, while towards the end, both are almost the same. The flat areas where zero order lines are picked correspond to when there is no activity within the warehouse. These comparisons show that the item generation in the model and the releasing and picking of orders are verified. The model does not spawn or delete more items than it is supposed to. This item generation and deleting verification and the order lines picked per hour, combined with the corrections during model creation, leads to the conclusion that the model is verified to simulate a replica of the as-is state of LS HMN. Further, multiple simulation runs have been conducted with all the possible configurations to ensure that the model is verified for its purpose and that all the different policies behave as planned.

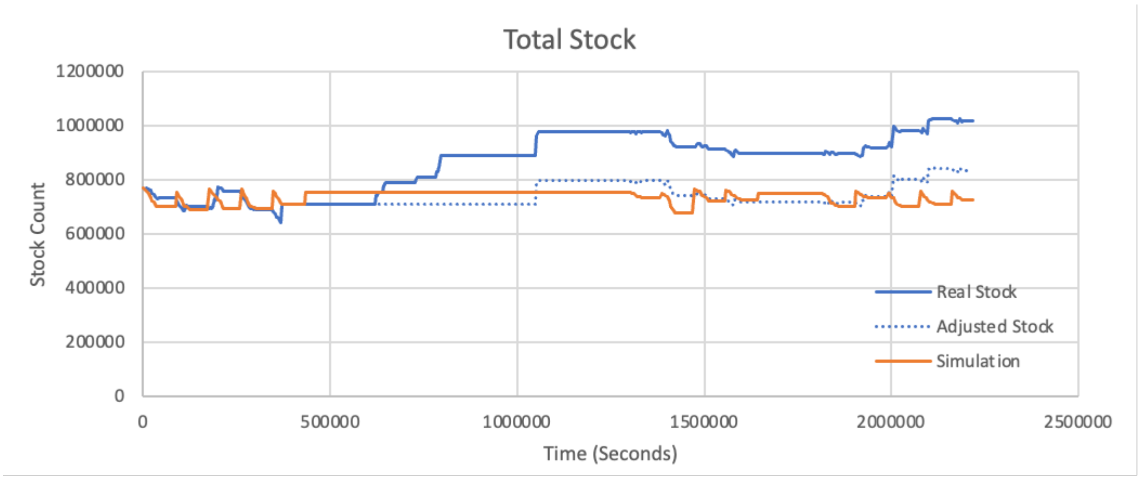


Figure 5.8: Verification and validation of stock level

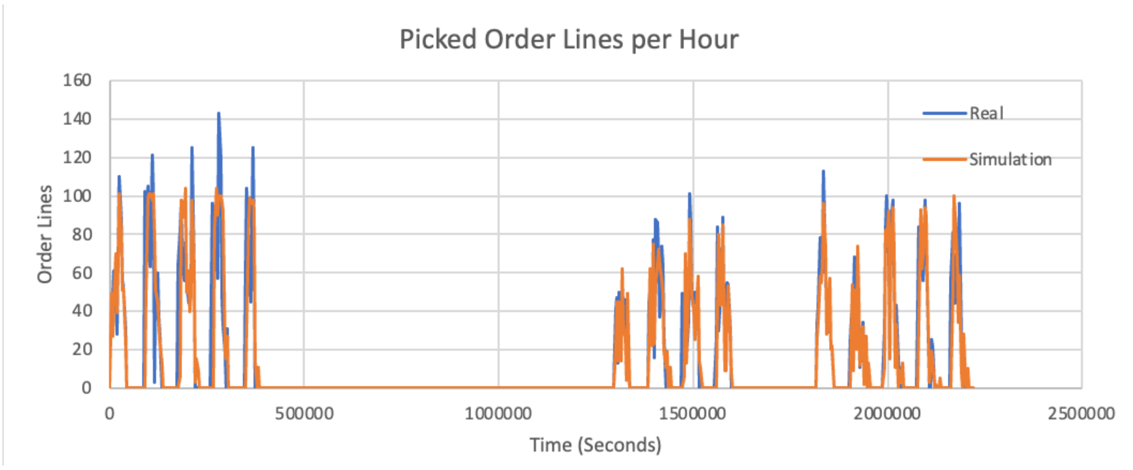


Figure 5.9: Verification and validation of picks per hour

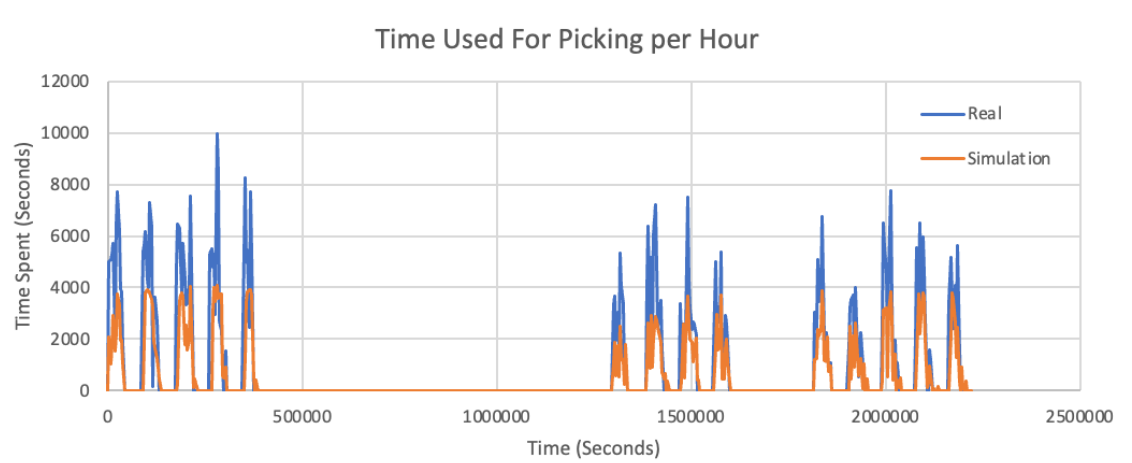


Figure 5.10: Verification and validation of time used for order picking

It was decided early in the development process that the model's accuracy does not have to be identical to the real system, as long as it was accurate enough to capture the differences in the scenarios, as stated by Sargent (2020) and Kleijnen (1995). Many of the model's parameters are extracted from the real system, as explained in Section 5.3, to make the model as accurate as possible. The most inaccurate part of the model is the picking time of an order or item since it is difficult to model the randomness of the human aspects. This difference is clearly shown in Figure 5.10, where the simulation spends around 50% less time on order picking than in the real system. In the data given by LS HMN, some operators spent almost 10 minutes picking an item, probably due to the operator taking a break. In the model, picking this exact order line takes as long as any other picking line.

The odd picking time happens multiple times in the real system, which is why the model spends way less time on order picking, as shown in Table 5.4. Therefore, the average time it takes for the model to process an order or pick an item is lower than the real system. Also, the picking time in the model has a lower coefficient of variation, meaning that the picking time variation is lower in the simulation than in the real system. The Mean Absolute Percentage Error (MAPE) between the simulation and real data is 44%. When the model processes one order, it will, on average, be processed 44% quicker or slower than within the real system.

Table 5.4: Actual and simulation order picking time comparison

	Actual	**Simulation**
Total time used for order picking	560 600	307 438
Average picking time per order	387	212
The standard deviation of picking time per order	344	156
Coefficient of variation	0.888	0.736
Average picking time per item	66	39
The standard deviation of picking time per item	59	22
Coefficient of variation	0.889	0.563
MAPE Order picking time		44%

The fixed model parameters for the operators, see Table 5.2, could be tweaked to make the model more realistic. However, this is time-consuming and a completely accurate model is unnecessary, as mentioned earlier. If the picking speed is increased or walking speed is decreased to make the model a more realistic replica of the real system, the waiting time for the operators might be reduced, and the policy changes might not be shown. If the waiting time is reduced, the VLMs would already be optimized for order picking because then the operators will be the constraint. This system's constraints are a mix of operators and VLMs. Therefore, changing the policies will affect the operators during order picking and how the different VLMs operate. Based upon this, the model is validated, and it is as accurate and simple as needed in order to serve its purpose, as stated by Sargent (2020).

5.6 Scenarios

To answer RQ 2 and 3, 48 scenarios are run where the policies are changed. When these scenarios are run, the amount of operators is set to two, and the active VLMs parameter is set to three. This configuration compares the current state at LS HMN to other configurations. The list of these 48 scenarios is shown in Table 5.5. Not all possible configurations and combinations uncovered and discussed in Section 4.2 are run. For instance, combinations with the two-bin system replenishment policy and the dedicated and OOS storage policies are not examined.

Firstly, dedicated storage requires that all the SKUs are given their own locations, which is a time-consuming operation in Flexsim®, and it can be discussed if the policy will be optimal. For LS HMN, the operators are given all the needed information about the item they are supposed to pick or replenish, including its location, eliminating the need to know exactly where items are stored. Also, the dedicated storage policy has multiple disadvantages, as stated in Section 3.1.3 and Section 4.2.3. Further, the OOS requires a complicated algorithm to be constructed, and the time it takes to assign storage locations might be long, depending on the number of families. Therefore, these storage policies are not examined because of the time constraints of this thesis and the disadvantages of the dedicating storage slots to each SKU. The same is applied to why the two-bin replenishment system is not examined.

The last 25 scenarios are run to answer RQ 4, are based on wishes of LS HMN. To assess what LS HMN can do to reduce order picking time within short terms and without doing drastic changes to their system, multiple scenarios are run where the amount of active VLMs and operators are changed. For LS HMN, changing the active VLMs parameter and adjusting the number of operators are simple changes. However, changing the other warehouse policies is a costly and time-consuming operation. Therefore, 25 scenarios are run where the number of operators and active VLMs is changed. These scenarios are shown in Table 5.6. The current policies applied at LS HMN and other applicable policies were introduced and discussed in Section 4.2, and the results of this discussion are shown in Table 4.5. The warehouse policies used in these first 25 scenarios are the same as the current policies in the real system, see the column named LS HMN in Table 4.5.

Table 5.5: Change of warehouse policies

Scenario	Replenish Policy	Replenish Level	Put-Away	Storage	Order Picking
1	FOI	VLM	No	Random	Discrete
2	FOI	VLM	No	Random	Batch (Up to 5 orders)
3	FOI	VLM	Discrete	Random	Discrete
4	FOI	VLM	Discrete	Random	Batch (Up to 5 orders)
5	FOI	VLM	Batch (Up to 10 items)	Random	Discrete
6	FOI	VLM	Batch (Up to 10 items)	Random	Batch (Up to 5 orders)
7	FOI	VLM	No	CBS	Discrete
8	FOI	VLM	No	CBS	Batch (Up to 5 orders)
9	FOI	VLM	Discrete	CBS	Discrete
10	FOI	VLM	Discrete	CBS	Batch (Up to 5 orders)
11	FOI	VLM	Batch (Up to 10 items)	CBS	Discrete
12	FOI	VLM	Batch (Up to 10 items)	CBS	Batch (Up to 5 orders)
13	FOI	Total	No	Random	Discrete
14	FOI	Total	No	Random	Batch (Up to 5 orders)
15	FOI	Total	Discrete	Random	Discrete
16	FOI	Total	Discrete	Random	Batch (Up to 5 orders)
17	FOI	Total	Batch (Up to 10 items)	Random	Discrete
18	FOI	Total	Batch (Up to 10 items)	Random	Batch (Up to 5 orders)
19	FOI	Total	No	CBS	Discrete
20	FOI	Total	No	CBS	Batch (Up to 5 orders)
21	FOI	Total	Discrete	CBS	Discrete
22	FOI	Total	Discrete	CBS	Batch (Up to 5 orders)
23	FOI	Total	Batch (Up to 10 items)	CBS	Discrete
24	FOI	Total	Batch (Up to 10 items)	CBS	Batch (Up to 5 orders)
25	ROP	VLM	No	Random	Discrete
26	ROP	VLM	No	Random	Batch (Up to 5 orders)
27	ROP	VLM	Discrete	Random	Discrete
28	ROP	VLM	Discrete	Random	Batch (Up to 5 orders)
29	ROP	VLM	Batch (Up to 10 items)	Random	Discrete
30	ROP	VLM	Batch (Up to 10 items)	Random	Batch (Up to 5 orders)
31	ROP	VLM	No	CBS	Discrete
32	ROP	VLM	No	CBS	Batch (Up to 5 orders)
33	ROP	VLM	Discrete	CBS	Discrete
34	ROP	VLM	Discrete	CBS	Batch (Up to 5 orders)
35	ROP	VLM	Batch (Up to 10 items)	CBS	Discrete
36	ROP	VLM	Batch (Up to 10 items)	CBS	Batch (Up to 5 orders)
37	ROP	Total	No	Random	Discrete
38	ROP	Total	No	Random	Batch (Up to 5 orders)
39	ROP	Total	Discrete	Random	Discrete
40	ROP	Total	Discrete	Random	Batch (Up to 5 orders)
41	ROP	Total	Batch (Up to 10 items)	Random	Discrete
42	ROP	Total	Batch (Up to 10 items)	Random	Batch (Up to 5 orders)
43	ROP	Total	No	CBS	Discrete
44	ROP	Total	No	CBS	Batch (Up to 5 orders)
45	ROP	Total	Discrete	CBS	Discrete
46	ROP	Total	Discrete	CBS	Batch (Up to 5 orders)
47	ROP	Total	Batch (Up to 10 items)	CBS	Discrete
48	ROP	Total	Batch (Up to 10 items)	CBS	Batch (Up to 5 orders)

Table 5.6: Operators and VLM scenario parameters

Scenario	Amount of Operators	Active VLMs
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	2	1
7	2	2
8	2	3
9	2	4
10	2	5
11	3	1
12	3	2
13	3	3
14	3	4
15	3	5
16	4	1
17	4	2
18	4	3
19	4	4
20	4	5
21	5	1
22	5	2
23	5	3
24	5	4
25	5	5

5.7 Summary

A detailed description of the DES model is given within this chapter. Firstly, the logic of the model is shown in detail for each of the main process flows. These process flows include the initial stock of the model, the order releasing and order picking flows, the item generation, and the VLM logic. Then the imported and exported data were mentioned. The imported data include the initial stock, orders and item info, which was all given from LS HMN, while the exported data includes the order picking time for each order line, the stay time of orders, the stock of the system per hour and the deleted order lines. Further, the parameters used within the model are listed. The parameters are divided into three groups; fixed parameters for the VLMs, fixed parameters for the operators, and variable parameters which change from scenario to scenario. Then, the limitations and the simplifications of the model are listed, before the model is verified and validated. In the end, the different simulation scenarios ran are motivated and listed.

Chapter 6

Results

Multiple scenarios have been simulated in the DES model and the results are shown in this chapter. Firstly, graphical representations of all the scenarios are shown. Then follows a brief mention of the effects of executing replenishment and put-away in parallel with order picking. The third part is an analysis of the effect on order picking time when changing warehouse policies. After this, multiple multivariate regression analyses are run to first underline the effect of executing replenishment and put-away in parallel with order picking, and then to underline the effects of changing warehouse policies. Lastly, the results from changing the active VLMs parameter and number of operators are shown graphically.

6.1 Effect of changing warehouse policies

In this section, the effects of changing the warehouse policies are shown. The complete list of data used in the following graphs and tables are shown in Appendix B. The values in each table in this section are the averages of the corresponding policies of each column in Appendix B. Further, the standard deviations shown are calculated from the values also given in Appendix B.

6.1.1 Graphical representations of each scenario

Changing the combination of warehouse policies affects the order picking time and replenishment order execution time. The following graphs show the difference between the 48 scenarios where warehouse policies are altered. The grey columns are when replenishment and put-away are not executed, and the blue columns are when replenishment and put-away are executed in parallel with order picking. Appendix C shows the same graphs but with different legends to show which scenario had which policy and to illustrate the effect of changing to a specific policy. It is clear that changing the combination affects the average time for picking one order and the average wait during order picking, see Figure 6.2. Further, the replenishment order execution time and waiting time are also subjected to changes when the combinations change, shown in Figure 6.3. Lastly, the stay time of both orders and replenishment orders are also affected when the combination of policies change, see Figure 6.4.

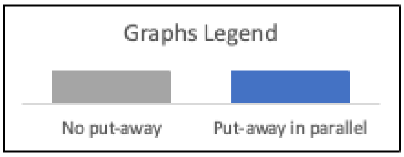
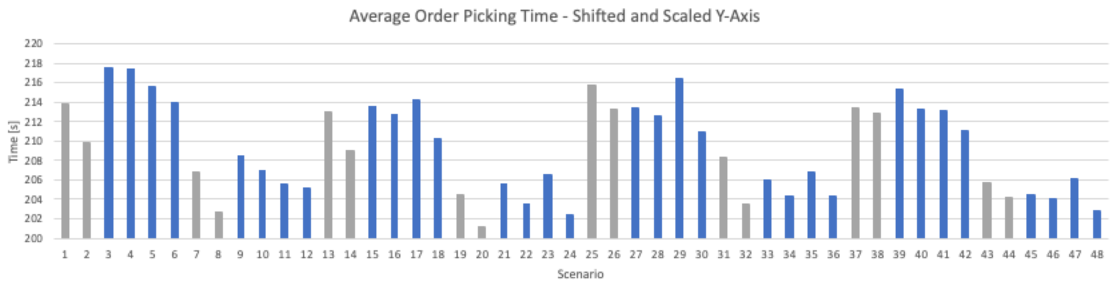


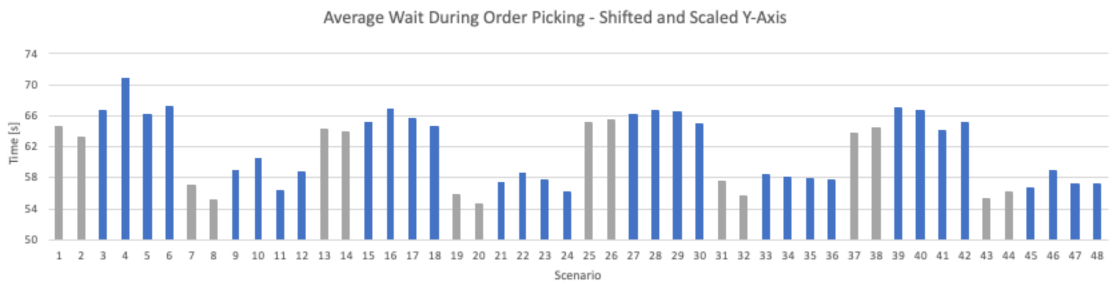
Figure 6.1: All scenarios legend

(a) Average time for picking one order



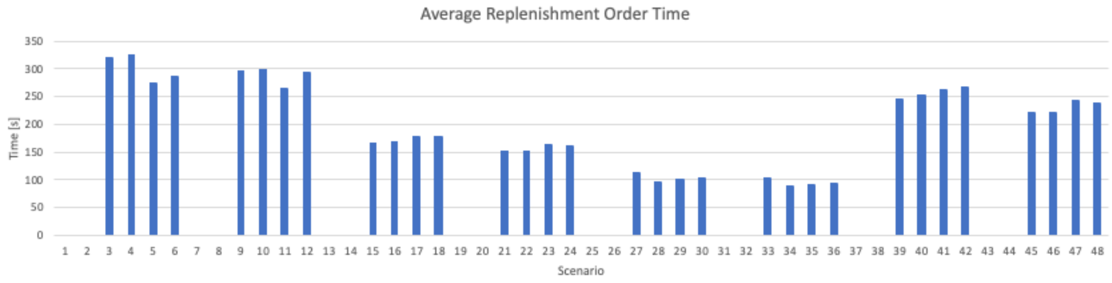
(b) Average time for picking one order - Shifted and scaled Y-axis

(c) Average wait during order picking

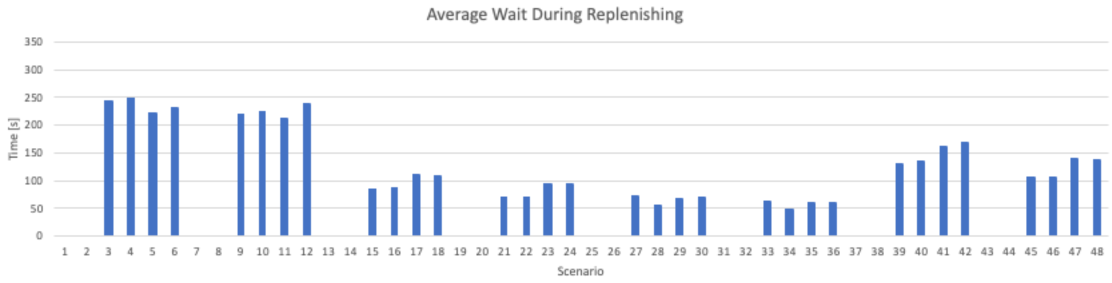


(d) Average wait during order picking - Shifted and scaled Y-axis

Figure 6.2: Average time to pick one order and average wait during order picking.



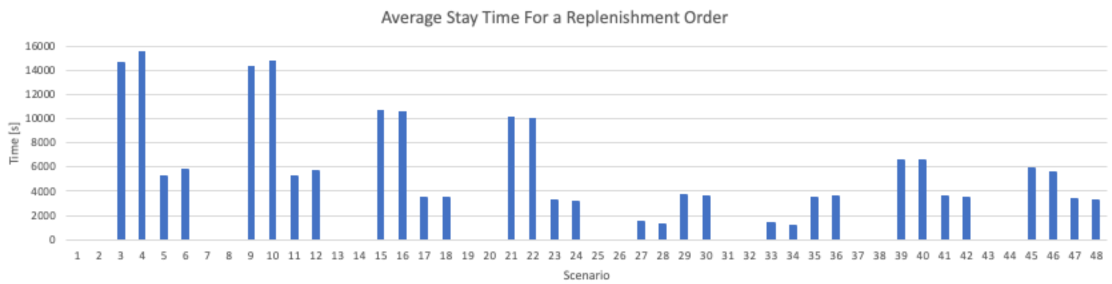
(a) Average time for executing one replenishment order



(b) Average wait during replenishment

Figure 6.3: Average time to execute one replenishment order and average wait during replenishment.

(a) Average stay time for an order



(b) Average stay time for a replenishment order

Figure 6.4: Average stay time for an order or a replenishment order.

6.1.2 Replenishment

When replenishment is executed in parallel with order picking, it does not affect order picking time or the waiting time during order picking by much, see Table 6.2a and Table 6.2c. The picking time is roughly the same, but it increases the average waiting time by around 3%. The slight increase in average waiting time could be due to the model's randomness or the changes in other policies. However, the replenishment affects the stay time for an order, see Table 6.2g. The stay time of an order is higher when replenishment is included because of the sequencing of activities are based on a FIFO logc, as mentioned in Section 4.1.3. When the order is released, there might be a queue of replenishment orders that must be executed before the order is picked because these replenishment orders were released before the order.

Both the level and the policy affect the system regarding replenishment. Firstly, the amount of replenishment orders changes between policy and level, see Table 6.1. The amount of replenishment orders is the same when using FOI at both the examined levels. However, when ROP is applied, the level affects the number of orders. When the level is set at the VLMs, there are more than twice as many replenishment orders as if the level is set at the total. Further, the scenarios with the quickest replenishment time, which also has the lowest waiting time and stay time, is when ROP is applied at the VLM level, see Table 6.2b, Table 6.2d and Table 6.2h. Contrary, the scenarios with the slowest average replenishment time, highest average wait during replenishment, highest average total time spent during replenishment and highest stay time are when FOI is applied at the VLM level.

It would have been interesting to investigate the order size of each replenishment order and see how it correlates to the rest of the data shown here, but the data about order size was not recorded. This is further discussed in Section 7.4.

Table 6.1: The amount of different replenishment order id's under different replenishment policies and levels

Policy \ Level	Nan	VLM	Total
FOI	-	751	751
ROP	-	1294	508

Table 6.2: Replenishment policy change results

(a) Average time for picking one order

Policy \ Level	Nan	VLM	Total
FOI	208	211	209
FOI - σ	*156*	*160*	*157*
ROP	210	209	209
ROP - σ	*158*	*158*	*157*

(b) Average time for executing one replenishment order

Policy \ Level	Nan	VLM	Total
FOI	-	296	166
FOI- σ	-	*163*	*94*
ROP	-	99	244
ROP - σ	-	*59*	*170*

(c) Average wait during order picking

Policy \ Level	Nan	VLM	Total
FOI	60	63	62
FOI - σ	*44*	*160*	*157*
ROP	60	62	62
ROP - σ	*45*	*45*	*45*

(d) Average wait during replenishment

Policy \ Level	Nan	VLM	Total
FOI	-	230	91
FOI - σ	-	*133*	*55*
ROP	-	62	136
ROP- σ	-	*43*	*96*

(e) Average total time spent during order picking

Policy \ Level	Nan	VLM	Total
FOI	297 844	302 772	299 246
FOI - σ	*6 714*	*7 587*	*6 496*
ROP	295 800	300 760	299 714
ROP- σ	*1 567*	*6 677*	*6 880*

(f) Average total time spent during replenishment

Policy \ Level	Nan	VLM	Total
FOI	-	225 114	125 927
FOI - σ	-	*16 737*	*7 626*
ROP	-	129 176	126 064
ROP - σ	-	*10 267*	*9 093*

(g) Average stay time for an order

Policy \ Level	Nan	VLM	Total
FOI	2 046	4 988	3 277
FOI - σ	*971*	*3 528*	*2 065*
ROP	2 031	2 414	2 482
ROP - σ	946	1 152	1 230

(h) Average stay time for a replenishment order

Policy \ Level	Nan	VLM	Total
FOI	-	10 179	6 873
FOI - σ	-	*6 972*	*5 105*
ROP	-	2 480	4 830
ROP - σ	-	948	5 165

6.1.3 Put-away

If replenishment is executed in parallel with order picking, put-away must also be executed in parallel. Put-away has the same effect on order picking as replenishment. The picking time, wait or total time used is barely changed, see Table 6.3a, Table 6.3c and Table 6.3e. However, the stay time is affected, see in Table 6.3g. The stay time of an order is increased due to the operators having more tasks to do. The average time to execute a replenishment order is also barely changed, see Table 6.3b. The slight change when replenishment orders are batched occurs because the operators do not have to walk to the pick-up point for each item. However, waiting for the VLM increases when replenishment orders are batched, see Table 6.3d. The increase is likely because the operators only walk to the next VLM and do not return to the pick-up location. Stay time for replenishment orders are more than doubled when the items are replenished discretely, see Table 6.3h.

Table 6.3: Put-away policy change results

(a) Average time for picking one order

	No	Discrete	Batch
Average	209	210	209
σ	*157*	*159*	*158*

(b) Average time for executing one replenishment order

	No	Discrete	Batch
Average	-	202	200
σ	-	*159*	*158*

(c) Average wait during order picking

	No	Discrete	Batch
Average	60	63	61
σ	*44*	*46*	*45*

(d) Average wait during replenishment

	No	Discrete	Batch
Average	-	123	136
σ	-	*84*	*94*

(e) Average total time spent during order picking

	No	Discrete	Batch
Average	296 822	301 389	299 857
σ	*14 749*	*7 029*	*6 495*

(f) Average total time spent during replenishment

	No	Discrete	Batch
Average	-	301 389	299 857
σ	-	*51 019*	*38 567*

(g) Average stay time for an order

	No	Discrete	Batch
Average	2 039	3 681	2 899
σ	*959*	*2 557*	*1 799*

(h) Average stay time for a replenishment order

	No	Discrete	Batch
Average	-	8 189	3 992
σ	-	*7 029*	*1 311*

6.1.4 Storage policy

The storage policy affects both the order picking and replenishment time. When CBS is applied, all the parameters are generally lower, see Table 6.4. All the order picking values, see Table 6.4a, Table 6.4c, Table 6.4e and Table 6.4g, is reduced by a few percentage points when CBS is applied compared to random. The same can be said about replenishing. All the replenishment values are lower when CBS is used compared to random, see Table 6.4b, Table 6.4d, Table 6.4f and Table 6.4h. The reduction occurs because to VLMs can extract the requested tray quicker for the items with a high turnover rate.

Table 6.4: Storage policy change results

(a) Average time for picking one order

	Random	CBS
Average	214	205
σ	*162*	*154*

(b) Average time for executing one replenishment order

	Random	CBS
Average	209	193
σ	*138*	*122*

(c) Average wait during order picking

	Random	CBS
Average	66	58
σ	*49*	*43*

(d) Average wait during replenishment

	Random	CBS
Average	138	122
σ	*96*	*82*

(e) Average total time spent during order picking

	Random	CBS
Average	306 698	294 5480
σ	*1 840*	*1 775*

(f) Average total time spent during replenishment

	Random	CBS
Average	158 155	144 986
σ	*4 379*	*4 209*

(g) Average stay time for an order

	Random	CBS
Average	3 389	3 192
σ	*2 255*	*2 166*

(h) Average stay time for a replenishment order

	Random	CBS
Average	6 262	5 919
σ	*5 250*	*4 854*

6.1.5 Order picking

The order picking policy does not affect the picking time by much, see Table 6.5a, Table 6.5c and 6.5e. However, the stay time of an order is increased by 280%, see Table 6.5g. The increase in stay time occurs because when the order is released and items are assigned, the timer of stay time starts, and then the orders are batched. Once up to five orders are released and batched, or until one hour after the first order arrived, the whole batch is released. If only one order arrives during one hour, the stay time of that order will start at 3600 seconds before it has even been released to the operator. Further, the order picking policy does not affect the replenishment time by much either, see Table 6.5b, Table 6.5d, Table 6.5f and Table 6.5h.

Table 6.5: Order picking policy change results

(a) Average time for picking one order

	Discrete	Batch
Average	211	209
σ	*159*	*157*

(b) Average time for executing one replenishment order

	Discrete	Batch
Average	200	202
σ	*130*	*131*

(c) Average wait during order picking

	Discrete	Batch
Average	62	62
σ	*46*	*46*

(d) Average wait during replenishment

	Discrete	Batch
Average	129	131
σ	*130*	*131*

(e) Average total time spent during order picking

	Discrete	Batch
Average	302 118	299 127
σ	*6 455*	*6 812*

(f) Average total time spent during replenishment

	Discrete	Batch
Average	151 467	151 673
σ	*42 661*	*47 702*

(g) Average stay time for an order

	Discrete	Batch
Average	1 975	4 605
σ	*2 459*	*1 932*

(h) Average stay time for a replenishment order

	Discrete	Batch
Average	6 061	6 120
σ	*5 011*	*5 103*

6.2 Regression analysis on the effect of changing warehouse policies

To further validate the results, multiple regression analyses have been run. First, multiple multivariate regressions have been run on the effect of implementing replenishment while doing order picking. Then, multivariate regression analyses were run on the effect of changing all the warehouse policies.

6.2.1 The effect of executing replenishment in parallel with order picking

The first four regressions, see Table 6.7, have been run to understand the effect of executing replenishment and put-away in parallel with order picking, especially how it affects order picking time. The grey rows indicate a statistical relationship between the current dependent variable and the policy change. The 32 scenarios where put-away is executed in batches and the replenishment level is set to total are removed to only look at the effect of implementing the different replenishment policies and discrete put-away. A total of 16 scenarios are within these regression analyses.

When replenishment is executed in parallel with order picking, only the stay time has a statistical relationship to the changes. This happens because the operators have more tasks to do. These results align with the previous analysis in Section 6.1, showing the same effects.

Table 6.6: Regression analyses legend

	0	1
Replenishment policy	FOI	ROP
Replenishment level	Nan	VLM
Put-away policy	Nan	Discrete

Table 6.7: Multivariate regression analysis on the effect of executing replenishment in parallel with order picking

(a) Regression analysis for average time for picking one order

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	210	2.23	94.0	< .001
Replenish	-0.84	2.58	-0.32	0.751
Put-Away	1.62	2.58	0.63	0.541

(b) Regression analysis for waiting during order picking

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	60.7	2.11	28.8	< 0.001
Replenish	-0.47	2.43	-0.19	0.850
Put-Away	2.80	2.43	1.15	0.271

(c) Regression analysis for total time

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	300 663	3 252	92.4	< 0.001
Replenish	39.0	3 755	0.01	0.992
Put-Away	1 924	3 755	0.51	0.617

(d) Regression analysis for average stay time of order

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	2 881	768	3.75	0.002
Replenish	-1 675	886	-1.89	0.081
Put-Away	2 134	886	2.41	0.032

6.2.2 The effect of changing warehouse parameters on order picking and replenishment

The order picking time is affected by multiple warehouse policies, see Table 6.9. The grey rows indicate statistical significance on the current variable. The replenishment level, storage policy and order picking policy have a statistically significant effect on the average order picking time, see Table 6.9a. When the replenishment level changes from VLM to total, the storage policy changes from random to CBS, and when order picking is executed in batches, the picking time is reduced. Further, the average waiting time is reduced by a couple of seconds when the replenishment policy and level, put-away policy and storage policy change from FOI to ROP, VLM to total, discrete to batch and random to CBS respectively.

When the replenishment policy, replenishment level and put-away policy change to, the total time used is reduced. The policies change from 0 to 1, see 6.8 As shown in Table 6.9d, changing the storage policy is the only variable that does not affect the stay time of an order. When orders are picked in batches, the average stay time is increased by around 57% due to the batching logic mentioned in Section 6.1.5.

The replenishment orders are affected by fewer policy changes than the order picking, but the effects are greater, see Table 6.10. The execution time for a replenishment order has only one statistically significant relationship, which is the replenishment policy. When the policy changes from FOI to ROP, the time is reduced by around 60 seconds. According to the rest of the regression analysis, the replenishment time has no statistically significant relationship with the replenishment level, the put-away policy or the storage policy. The same can be said about the waiting during replenishment, see Table 6.10b.

When it comes to the total time used for replenishment, see Table 6.10c, the time is significantly reduced by changes in the replenishment and put-away policies. Lastly, the average stay time is affected by both the replenishment policy and the replenishment level, see Table 6.10d. When the replenishment and put-away policies change, the average stay time is reduced by around 90%.

Table 6.8: Regression analyses legend

	0	1
Replenishment policy	FOI	ROP
Replenishment level	VLM	Total
Put-away policy	Discrete	Batch
Storage policy	Random	CBS
Order picking policy	Discrete	Batch

Table 6.9: Multivariate regression analysis on the effect of warehouse policy changes on order picking

(a) Regression analysis for average time for picking one order

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	217	0.57	384	< 0.001
Replenish	-0.91	0.46	-1.96	0.060
Replenish Level	-1.66	0.46	-3.609	0.001
Put-Away	-0.89	0.46	-1.92	0.066
Storage	-8.63	0.46	-18.73	< 0.001
Order Picking	-2.05	0.46	-4.45	< 0.001

(b) Regression analysis for waiting during order picking

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	67.3	0.44	153	< .001
Replenish	-0.52	0.36	-1.43	0.163
ReplenishLevel	-1.01	0.36	-2.808	0.009
Put-Away	-1.26	0.36	-3.51	0.002
Storage	-8.37	0.36	-23.36	< 0.001
OrderPicking	0.67	0.36	1.86	0.074

(c) Regression analysis for total used for order picking

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	310 488	733	423	< 0.001
Replenish	-772	599	-1.29	0.209
ReplenishLevel	-2 286	599	-3.82	< 0.001
Put-Away	-1 532	599	-2.56	0.017
Storage	-12 151	599	-20.29	< 0.001
OrderPicking	-2 991	599	-5.00	< 0.001

(d) Regression analysis for average stay time of order

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	3 718	264	14.07	< 0.001
Replenish	-1 684	216	-7.81	< 0.001
ReplenishLevel	-822	216	-3.81	0 < .001
Put-Away	-783	216	-3.63	0.001
Storage	-197	216	-0.91	0.369
OrderPicking	2 630	216	12.19	0 < .001

Table 6.10: Multivariate regression analysis on the effect of warehouse policy changes on replenishment order execution

(a) Regression analysis for average time for executing a replenishment order

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	234.56	33.41	7.02	< 0.001
Replenish	-59.01	27.28	-2.16	.040
ReplenishLevel	7.54	27.28	0.28	.784
Put-Away	-1.80	27.28	-0.07	.948
Storage	-15.41	27.28	-0.57	.577
OrderPicking	1.91	27.28	0.07	.945

(b) Regression analysis for waiting during replenishment

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	177.91	26.12	6.81	< .001
Replenish	-61.60	21.32	-2.89	0.008
ReplenishLevel	-33.12	21.32	-1.55	0.132
Put-Away	12.55	21.32	0.59	0.561
Storage	-15.99	21.32	-0.75	0.460
OrderPicking	1.76	21.32	0.08	0.935

(c) Regression analysis for total time used for replenishment and put-away activity

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	209 122	12 212	17.12	< 0.001
Replenish	-47 901	9 971	-4.80	< 0.001
ReplenishLevel	-51 150	9 971	-5.13	< 0.001
Put-Away	-3 090	9 971	-0.31	0.759
Storage	-13 169	9 971	-1.32	0.198
OrderPicking	206	9 971	0.02	0.984

(d) Regression analysis for average stay time of a replenishment order

Warehouse Policy	Coefficients	Standard Error	t	P-value (Sig.)
(Constant)	11 006	1 261	8.73	< 0.001
Replenish	-4 872	1 030	-4.73	< 0.001
ReplenishLevel	-478	1 030	-0.46	0.646
Put-Away	-4 197	1 030	-4.08	< 0.001
Storage	-345	1 030	-0.33	0.742
OrderPicking	60	1 030	0.06	0.954

6.3 Effect of varying the active VLMs parameter and number of operators

The change in the number of operators and active VLMs affects the system's performance quite drastically. The legend for the following graphs is shown in Figure 6.5. The black diamond is the same scenario as LS HMN, two operators and three active VLMs.

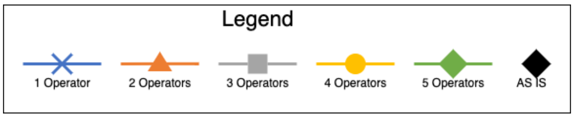
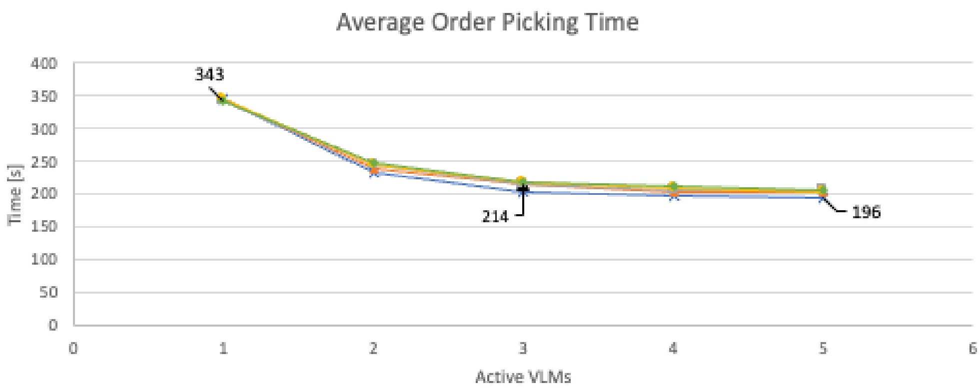


Figure 6.5: Change of active VLMs parameter and number of operators scenarios legend

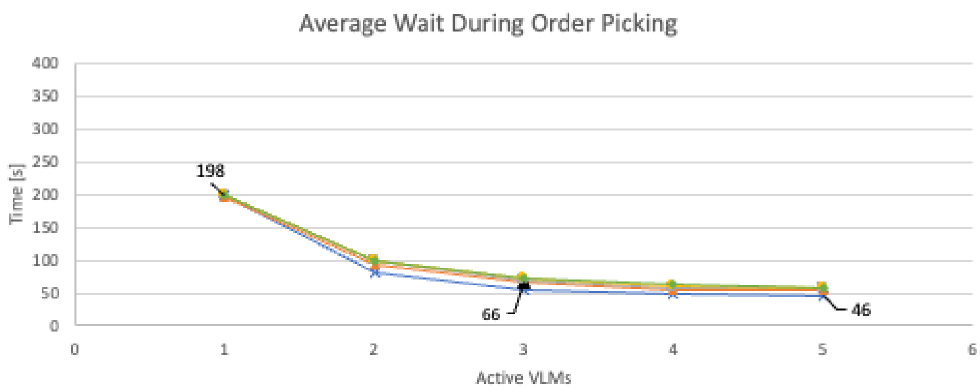
Firstly, the amount of operators does not affect the order picking time, see Figure 6.6a, nor the average wait during order picking, see Figure 6.6b. However, changing the parameter of active VLMs affects the order picking values drastically. When only one VLM is active, the order picking time is, on average, 75% slower than when the parameter is set to five. For LS HMN, increasing the parameter might reduce the order picking time by 16%. Regarding waiting during order picking, the effect is even more prominent when the parameter increases. The operator waits, on average, 4.3 times longer for the VLM when only one is active compared to when five are active. For LS HMN, the average waiting time during order picking can be reduced by 30% when changing the parameter to five. Further, when changing the parameter, the wait percentage per order is reduced from 58% to 23%. Reducing the wait time percentage means that the system bottleneck shifts from the VLMs to the operators.

The more operators used, the lower the average stay time for an order, see Figure 6.6c. When increasing the number of operators from one to two, the stay time is reduced by 80%. Comparing the worst scenario to the best, one operator and one VLM and five operators and five VLMs respectively, the stay time is reduced by 97%. For LS HMN, the stay time can be reduced by 72%, from 1715 seconds to 486 seconds per order.

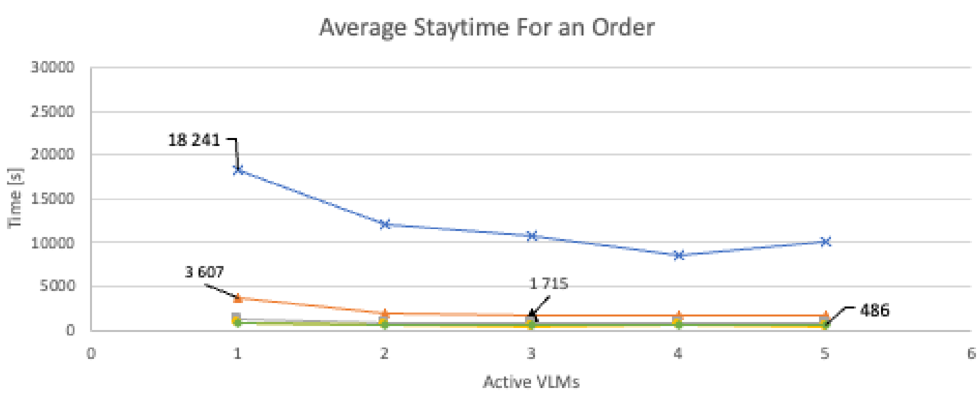
The replenishment time and wait during replenishment, see Figure 6.7a and Figure 6.7b, are more subjected to changes when the amount of operators changes, compared to order picking. Generally, the average replenishment time increases when the number of operators increases, except for when one and four operators are used. The configuration which leads to the lowest replenishment time and waiting during replenishment is when one operator is used, and five VLMs are active. The time for replenishment does not change as much when the active VLMs parameter is changed because the parameter is only applied to order picking. However, the parameter affects the stay time for replenishment, see Figure 6.7c.



(a) Average order picking time

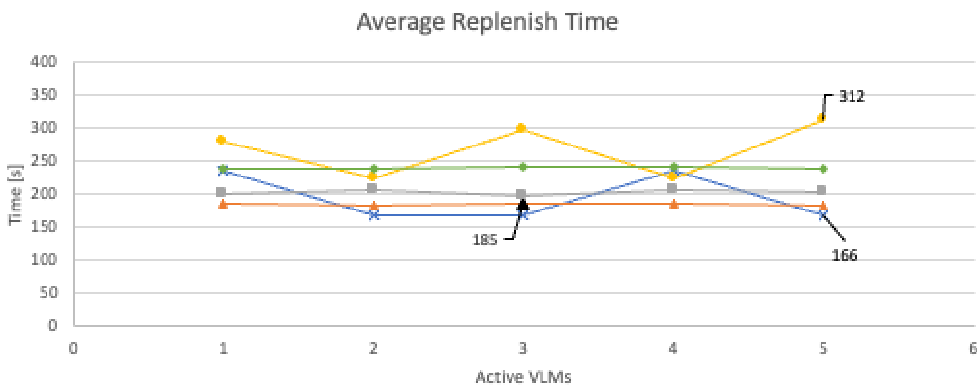


(b) Average wait during order picking

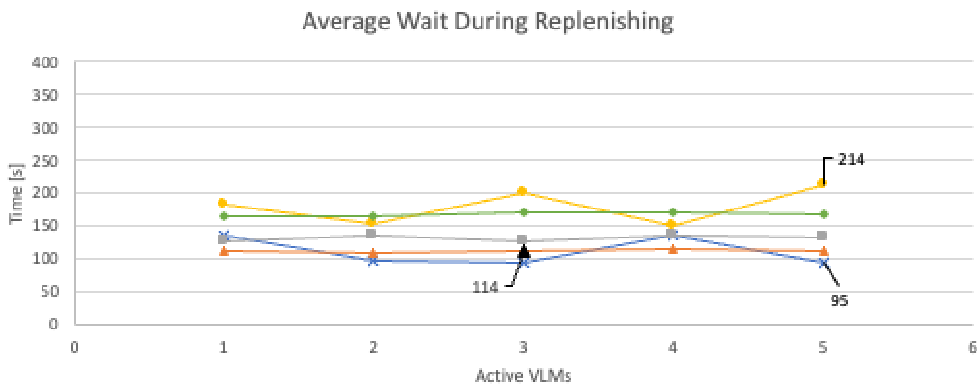


(c) Average stay time for an order in the model

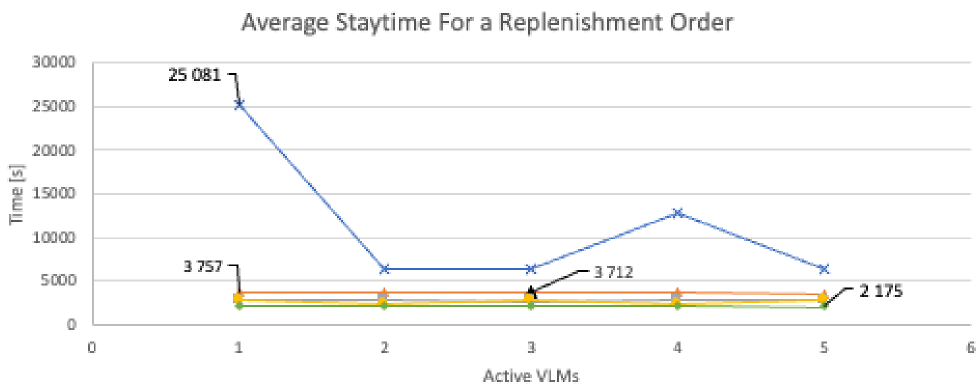
Figure 6.6: Varying the active VLMs parameter and number of operators scenarios - Order picking



(a) Average replenishment order time



(b) Average wait during replenishment



(c) Average stay time for a replenishment order in the model

Figure 6.7: Varying the active VLMs parameter and number of operators scenarios - Replenishment

Chapter 7

Discussion

This chapter begins with a discussion of the results. First is a discussion regarding how the replenishment and put-away activities affect the order picking when they are executed in parallel. Then follows a discussion about the results of the effect of changing warehouse policies. The last discussion regarding the results is when the active VLMs parameter and the number of operators are changed. The chapter ends with a discussion of limitations and criticism regarding the thesis and chosen methods.

7.1 Replenishment and put-away executed in parallel with order picking

When replenishment and put-away is executed in parallel with order picking, the average order picking time is not affected, as shown in both Section 6.1.2 and Section 6.2.1. The operators should not be affected in executing the picking or the time it takes to walk from one pick location to another. However, the waiting time might increase, see Table 6.2c. When more tasks should be executed, the operator might have to wait more often to complete their tasks because the VLMs are busier, especially if multiple operators are used. The increase in the number of operators is discussed in Section 7.3. Further, when there are more tasks, the queue of tasks will increase, resulting in a longer stay time for the orders. The effects of this are seen in Table 6.2g and Table 6.7d.

The average stay time of an order increases by a couple of hundred seconds when replenishment and put-away are included, but the amount depends on the current policy. The effect of the policy changes will be further discussed in Section 7.2. Table 6.7d shows that the sum of the coefficients of replenishment and put-away is positive, indicating an increase in stay time. In order to reduce the stay time, the sequencing and queuing of tasks could be reorganised to either prioritise order picking or replenishment.

To conclude, the order picking time is not directly affected when replenishment and put-away are executed in parallel with order picking. However, the stay time of an order is affected because more tasks should be executed, which is further affected by the sequencing of tasks. In a real system, the waiting time might increase if replenishment and put-away are executed in parallel with order picking. The number of tasks increases, and if multiple operators are used, the chance of waiting for other operators to finish is higher compared to when only one operator is used. Also, increasing the number of operators would increase the number of active VLMs, which would further increase the chances of waiting.

7.2 The effects of changing warehouse policies

7.2.1 Replenishment

The replenishment policy and replenishment level affect the number of replenishment orders. When FOI is used, the amount of replenishment orders stays the same no matter which replenishment level is used. The replenishment order amount stays the same because of the fixed time interval of when the stock is checked and because the output is always the same in each scenario. At each time interval that the stock is checked (once a day), the stock of each SKU within each scenario has dropped to the same level, no matter the replenishment level. This is because the orders are released simultaneously between each scenario, and the orders are identical between the scenarios. However, the number of replenishment orders when ROP is used differs between the two replenishment levels examined.

When the replenishment level is set to individual VLMs and ROP is used, the number of replenishment orders is almost three times as big as when the level is set to total. The increased number of replenishment orders occurs because the reorder point is reached more often when individual VLMs are used compared to the total. Intuitively, the replenishment order sizes are smaller when the level is set to VLM, no matter the replenishment policy. However, this data was not extracted from the model. This is further discussed in Section 7.4.2. The differences between replenishment policy and level affect both the order picking time and the replenishment order execution time, see Table 6.2 and Section 6.2.2.

According to the regression analysis for order picking time, Table 6.9, the order picking time, waiting time, and stay time are all affected by the replenishment level. When the level changes from VLM to total, the values are decreased. These results correlate with the ones shown in Table 6.2. However, according to Table 6.2g, the stay time of an order is ever slightly increased when ROP is used with the level set to total compared with VLM. Besides this increase, the order picking values are generally lower when the level is set to total compared to individual VLMs. The difference in replenishment policy only affects the stay time of an order. It does not affect any of the other order picking values investigated. When FOI is used, the average stay time of an order is higher compared to ROP.

Further, the replenishment order execution time is more subject to changes when both the replenishment policy and level change, see Table 6.2. The configuration with the longest average executing time, waiting time, total time used, and longest average stay time is when FOI is used at the individual VLM level. Contrary, the configuration with the lowest of these examined values is when ROP is used at the VLM level. Exactly why these are the configurations with the highest and lowest execution time is difficult to answer, especially without the average replenishment order size. The order size and frequency should not differ between the two levels when FOI is used because of the same output and order interval between each configuration. The time difference between the ROP levels is due to the order size. When the level is set to total, the ROP is reached less often but with a bigger replenishment order size. The bigger replenishment order size occurs because each scenario has the same output, but the amount of replenishment orders differ.

7.2.2 Put-away

The next warehouse activity examined is the put-away activity. The put-away can be executed discretely or in batches, and the two policies affect order picking and replenishment. Firstly, according to both Table 6.3 and the regression analysis Table 6.9, the order picking time is not affected by changing the put-away policy, but the waiting time is slightly decreased from discrete to batch. The decrease in waiting time should also be reflected in the order picking time because the waiting time is a part of the order picking time. Arguably, if the wait during order picking is affected, the order picking time is also affected.

Further, the total time used is reduced when the average order picking time is reduced. Lastly, when the policy changes from discrete to batch, the stay time for an order is decreased by 21%, shown in both Table 6.3g and Table 6.9d. The stay time is reduced because when the items are batched, they are picked up and unloaded quicker than when a single item is picked up. In the model, the time it takes to pick up a replenishment order is the same no matter the size, and therefore the stay time of a replenishment order is reduced, which further reduces the order stay time. Also, when items are batched, the operator does not have to return to the input area between each unloading. To follow up on the replenishment order, according to the regressions in Table 6.10, the only replenishment variable that is affected by changes in put-away is the stay time for a replenishment order.

Changes in the put-away policy affect the stay time for a replenishment order. When batched put-away is used, the stay time for a replenishment order is 51% less compared to discrete put-away. This reduction is because the operator does not have to go to the input area between each put-away and because of the model's logic, as just mentioned. Further, according to Table 6.3, the average wait during replenishment is increased when batch put-away is used, but the execution time is not affected. This is because the travelling time is reduced when batch is used, but the time it takes for a VLM to display a tray is the same. The difference in waiting time only occurs because the operator walks less when batch is used. Therefore, some of the waiting time when batch is used is swapped with walking time when discrete put-away is used.

7.2.3 Storage

The following warehouse policies examined are the two different storage policies. The two policies show a significant difference, shown in Table 6.4 and Table 6.9. When random is used, the order picking time, wait during order picking, total time used for order picking and the average stay time for an order is higher than when CBS is used. The same can be seen for the replenishment orders. The execution time, waiting time, total time used, and stay time is lowest when CBS is used. This is in line with the conclusion of Daria et al. (2015), the time spent for order picking is lower when CBS is used compared to random. Intuitively, the same should happen with the replenishment orders, as shown. The time spent when CBS is applied is generally lower because around 80% of the items picked are extracted quicker by the VLMs.

In these results, the CBS is configured to classify the different trays. However, the classification can also be between the different VLMs. If applied to the different VLMs, some VLMs will be visited more often than others, and the waiting time will potentially increase. The increase in waiting time would increase the order picking time and the stay time for an order, which would affect the system negatively. This applies to both order picking and replenishment order execution. The waiting time will increase because the probability that two or more consecutive order lines are stored in the same VLM increases, especially for the items with the highest turnover. This configuration could be implemented, but the effects will be adverse. In addition, the VLMs with the items with the lowest turnover rate would seldom be used.

7.2.4 Order picking

In the results chapter, it is shown that the order picking policy only affects the system a little. When batch picking is used, the order picking time is reduced by a minuscule amount, and the waiting time during order picking is increased by a single second, compared to discrete order picking. However, the stay time of an order is more than doubled when batch is used. The increase in order stay time occurs because after the items are assigned to an order, it waits until five orders are batched, up to a maximum waiting time of one hour, before the order is released. The stay time is recorded during this process, as mentioned earlier.

Further, the replenishment orders are affected in the same way. The waiting time is bearly increased, and the stay time of a replenishment order is increased. However, the replenishment order execution time is increased compared to the order picking time when batch picking is used. The slight increase occurs because of the waiting time for the orders and replenishment orders. These results are seen in Table 6.5, Table 6.9 and Table 6.10.

The effect of changing between discrete and batch picking is not so substantial when VLMs are used. Within a VLM, it is not given that multiple items are stored at the same location (VLM or tray), as in a warehouse where the forward picking area often is located in the first levels of a rack. If an order request two full boxes, the operator still probably has to walk between picking the two boxes when VLMs are used. Therefore, batching orders might not have the same impact on order picking time when VLMs are used compared to if racks are used. The most considerable effect of batching orders is that the operator does not need to walk to the same location to pick a specific SKU repeatably.

The replenishment policy does not affect the order picking time or wait, only the stay time. When ROP is used, the stay time of an order is the lowest. Further, the replenishment level affects both the order picking time and the average stay time of an order. If the replenishment level is set to total, the order picking time, wait during order picking and stay time are lower than if the level is set to individual VLMs. Further, the put-away policy barely affects the order picking time and the replenishment execution time. Both are executed a fraction quicker when batch is used compared to discrete. However, when batch is used, the stay time for an order is reduced by 21%, and it is reduced by 51% for a replenishment order. The decrease occurs because the operators walk less, and the pick-up time for a replenishment order is the same, no matter the number of items.

Further, the change in storage policy from random to CBS leads to decreased order picking time and replenishment execution time. As a result, the average stay time for both orders and replenishment orders is also reduced. For order picking, there is not a big difference between discrete and batch picking, according to the results. When batch picking is used, the order is picked a minuscule amount quicker, and the waiting time is increased by a second, but the stay time is more than doubled compared to discrete order picking.

7.3 The effect of varying the active VLMs parameter and the number of operators

When the active VLMs parameter and the number of operators are changed, the system reacts differently between order picking and replenishing. For order picking, the time used will be lower when the parameter is set to a higher integer, while the number of operators does not affect the time as much. Contrary, the replenishment and put-away execution time is not affected by changes in the parameter, but the number of operators has an impact.

For order picking, the change in the active VLMs parameter reduces the average order picking time because the operators generally have to wait for less on the VLMs. The walking and the activity of picking the item are not affected because changing the parameter does not affect how fast the operators walk or pick one item. The wait time is reduced because while the operator picks an item from a tray, the next VLM(s) can fetch the next tray(s), and when the operator is at the next VLM, the trays are already displayed, reducing the needed waiting time. The effect of increasing the parameter is the highest when changing the parameter between 1 and 2. The decrease in picking time, primarily because of the reduction in waiting time, is most significant between these because the system goes from not readying any other order lines than the current to reading the following order line. Increasing the parameter will gradually reduce the order picking and waiting time. The higher the parameter is set, the lower the effect of increasing the parameter.

The decrease in effect by increasing the parameter occurs because the correct tray will always be displayed and ready to be picked from when the operator arrives at the VLM. If the VLMs always displayed the correct tray, the waiting time would be 0, and the constraint of the system would be the operators walking and picking time, not the time it takes for a VLM to fetch and display a tray. However, the waiting time does not go towards 0, but it flattens out at around 40 to 50 seconds, as shown in Figure 6.6b. The waiting time flattens out because, quite occasionally, two consecutive items are stored within the same VLM. When this happens, the operator has to wait for the VLM to store the current tray and fetch the next one. To further reduce the waiting time, the warehouse policies can be set to a configuration that lowers the time it takes to fetch a tray. For instance, using CBS will use less time to store and extract a tray compared to using a random storage policy, as shown by Daria et al. (2015) and in Table 6.4.

To further reduce the waiting time and the average order picking time, the WMS, which assigns the items to the orders, could be altered to not assign two consecutive items in the order from the same VLM. Not assigning two consecutive items from the same order to the same VLM would reduce the need for operators to wait at the same VLM. However, if this is included, the operators might have to wait more for the other operators if multiple operators are used. It is clearly shown that changing the number of operators does not affect the model's order picking time or waiting time. This is due to the way the tasks of the operators and the VLMs are queued, which is discussed in Section 7.4.1. However, if the waiting time is reduced to a minimum, increasing the number of operators might affect the waiting time because operators might have to wait for the other operators to finish their tasks. The waiting for other operators event is something LS HMN have experienced, but this thesis does not discuss it.

Replenishment orders are also affected by changes in the number operators and the active VLMs parameter. The replenishment order's execution and waiting times are primarily subject to changes in the number of operators. When the number of operators increases, the average time increases. The increase is due to more waiting on the other operators when the number of operators increases, especially when the put-away is executed discretely and not batched. The replenishment time might be reduced if the put-aways are executed in batches.

There are a couple of outliers in the results in Figure 6.7a and Figure 6.7b. When the active VLMs parameter increases, values for one and four operators change. This change is not in line with the other results within the graphs. Comparing these two graphs with the other three graphs, it can be argued that when one operator is used, the values should be lower than when two operators are used. If four operators are used, the values should be between three and five operators. If this is true, increasing the number of operators should increase the replenishment time, especially when the put-away is executed discretely.

Although changing the number of operators does not affect the picking time or the waiting time within the model, it affects the stay time for both an order and a replenishment order. When the number of operators is increased from 1 to 2, the stay time of an order is significantly reduced, shown in Figure 6.6c. The reduction occurs because more operators can execute the order picking or put-away tasks. The stay time of orders and replenishment orders are mainly subjected to changes when the number of operators is altered. However, it is also affected when the active VLMs parameter is altered. The stay time is reduced because the average order picking time is reduced, reducing the stay time for replenishment orders. However, the parameter is not directly linked to the replenishments.

In conclusion, increasing the number of operators does not affect the order picking time or the waiting time during order picking in the model. However, in a real system, the waiting time might increase. Further, increasing the active VLMs parameter will drastically reduce the order picking time and the waiting time during order picking. The time reduction of setting the parameter to five instead of one reduces the order picking and waiting time by 42% and 76%, respectively. Further, replenishment and put-away execution time is not affected by changes in the active VLMs parameter.

However, when the amount of operators increases, the execution and waiting time increase. Lastly, the stay time of both orders and replenishment orders is significantly reduced when two operators are used instead of one. The stay time is lowered when more operators are used, but the effects on the stay time decrease when the number of operators increases.

7.4 Limitations

During the development of this thesis, multiple simplifications were implemented, and multiple limitations are uncovered. Firstly, multiple warehouse policies are mentioned and discussed but not simulated and analysed. Further, multiple simplifications and limitations within the DES model need to be highlighted and discussed. Lastly, limitations within the data processing and missing data need to be commented on.

Multiple warehouse policy combinations are mentioned but not analysed in this thesis. Combinations where the *two-bin system* replenishment policy and *OOS*, *dedicated* and *CBS between VLMs* storage policies are used is not analysed. To check all combinations, a total of 180 scenarios must be simulated. Further, the dedicated storage has unlimited configurations available, and the batch put-away or order picking limit can be changed. The most optimal configuration of these is subject to specific cases.

7.4.1 The discrete event simulation model

Although the model was validated and verified, it has multiple flaws and limitations. Some were implemented purposefully and therefore mentioned in Section 5.4, and some were discovered during the data analyses. Unfortunately, these were neither fixed nor included due to time limitations in developing this thesis. However, these flaws must be discussed to understand how they have affected the result and how they could have been avoided.

Firstly, the limitations and simplifications mentioned need to be further discussed. The simplification that all items have the same size reduced the amount of imported data and eased the development process for storing the items. When all the items have the same size, the dimensions of the slots within each rack be resized to fit up to 12 items, which enables each level only to have one slot. This simplification helps the model find items. The model needs the specific location of an item in order to find the rest of the items stored on the same level. During the development of the model, this logic was easier to implement if all items were stored in the same slot, than if each tray had 12 slots and each item could have different sizes. Further, the sizes do not matter within the model but in the real system.

In the real system, when trays are stored, the tray height is scanned to utilise storage height and volume best. This logic is not included due to the increase in model complexity. In Flexsim®, the racks have fixed positions for the levels. Although all the rack dimensions, including levels and slots, can be altered, the rack is unchangeable during the simulation. In order to include the dynamic positioning of trays, the logic of the racks has to be reprogrammed.

Further, the thesis does not try to optimise the storage volume in any way, only changes in the warehouse policies. Therefore, spending time reprogramming a small part of the logic was deemed unnecessary. The outcome of including it might have decreased the picking time by a couple of seconds because the crane does not need to travel as high for the upper levels, especially when a random storage policy is used. The probability of needing an item from the upper levels is higher when random storage is used compared to CBS.

Another limitation of the model's logic is that only one item can be picked from a tray. If multiple items are supposed to be picked from the same tray, the tray has to go into storage and be displayed again between each pick. In the real system, the tray would be displayed until all the items were picked. Multiple attempts were taken to implement this logic during the development of the model. However, all the attempts failed due to how Flexsim® handles the information regarding items currently stored and those that are temporarily not stored. Because this logic was not included, the operators might have to wait longer during order picking or replenishment order execution. However, the probability that the same tray is requested twice is low when the same SKU is stored in multiple locations. To avoid this, the logic could have been altered to not find items within the same tray as the prior item in the order.

Further, a logic regarding deleting items from orders was included. Occasionally, when an item was moved back into the racks, the item's state was not changed to stored, and therefore the model could not find the item when searched for. The find items logic implemented by Flexsim® can only locate items that whose state is stored. When items state are not changed back to stored, the item cannot be located. When this happens, the model does not register that the item has disappeared. Therefore, the stock level is not updated when the item is "lost". Since the stock level is not updated, the model assumes there is enough stock and the ROP or FOI is not triggered, and so the item or SKU will cease to exist. If the SKU does not exist, the model cannot find it either, and so the order lines with these "lost" SKUs have to be deleted. If they are not deleted, the model will stop. During the simulation, less than 1% of all order lines were deleted because of this logic, and therefore it was included. Finding this bug in the model is time-consuming, and because of the scarcity of the event, the bug was not fixed. Fixing the bug would not affect the results in a significant way since it only happens ever so often.

Regarding the order picking process, four important aspects of the logic arise. The first is the order picking logic of spawning and deleting items, as mentioned in Section 5.4, the second is how the batching of orders works within the model, compared to the real system and the third aspect is the picking time of items and how changes in it would impact the results. The final aspect is how the operators and VLMs tasks are queued.

Firstly, the order picking logic of spawning and deleting items when operators pick them is included to enable the same flow item to be picked from repeatedly. When the item is assigned to an order, the quantity is reduced by the order amount or to 0 if the order amount is greater than the item's quantity. Then, when the operator is at the VLM, and the item is available for picking, the operator is stationary for a time corresponding to the time it takes to pick an item. When the timer is finished, an item is spawned in the hands of the operator, which corresponds to the item within the order. If the quantity of the item in the VLM is 0, the item is destroyed, and its data is deleted. Otherwise, if the quantity is greater than 0, the item is stored again. This logic was included to ensure that the software understood when items should be picked from multiple times and when the items should be destroyed. If the logic were not included, each item would be picked and removed from storage, no matter the order quantity or the item's quantity. Then the storage level of each SKU would not be accurate, which would stop the generation of replenishment orders, which would further stop the model.

Secondly, the batching of orders within the model is not as realistic as it should be. In a real system, when orders are batched, the order lines and items are reorganised so that the order lines with the same SKUs are consolidated and sequenced consecutively to reduce the walking needed during order picking. In the model, when orders are batched, the orders are just merged while still having the same sequence within the order. The second order is put straight after the first order without changing the order line sequence. Herefore, the effects on order picking time by changing the batching to be more realistic might not be significant. The only effect would be an increase in full boxes and a decrease in less-than-full-box picks, reducing the picking time by a small amount.

Further, the picking time for operators could affect the results. The picking time for one item is set to be normally distributed to make the model more realistic. However, this decision could have affected the results. If the picking time had been constant, the effects of changing warehouse policies could have been shown more clearly. The operators might have to wait less if it takes longer to pick an item than if it is picked quickly. The results would be approximately the same for the average order picking and replenishment execution time. However, the effects of changing warehouse policies on the waiting time for the VLMs would be more quantifiable.

Finally, the last part of the logic that needs to be discussed is the queuing of tasks for both the operators and the VLMs. At LS HMN, the queue is based upon FIFO for the VLMs, and one VLM can have a queue depending on the current tasks within the warehouse. In the model, the logic behind the queuing of tasks between the operators and VLMs have some flaws. The logic had to be included for the active VLMs parameter to function. Firstly, the queuing of operator tasks is only based on FIFO within the model. The token that first gets to the "Acquire: Operator" activity in Appendix A in both the

Order picking - Do batch picking and *Item generation - Replenish* boxes will acquire the operator first, no matter if there is a high demand of either orders or replenishment orders. The model will not prioritise tasks based on the demand of the tasks.

Further, the queuing of tasks for the VLMs have a flaw. This specific queuing logic is seen in Appendix A the *Order Picking - Pick Items* box within the section after the *"Which operator"* activity. Firstly, the *VLMs 01* activity limits the number of order lines to be executed simultaneously. The amount of order lines depends on the active VLMs parameter. Then comes the *aquire bay* activity. Here the token stays until the VLM it is supposed to request is available. This single *acquire bay* activity is needed to ensure that multiple orders do not acquire the same tray simultaneously. The token waits at this activity until the VLM is ready. This waiting will alter the sequencing of the order lines if the active VLMs parameter is any higher than one. The VLM for order line N+1 could be available before the VLM for order line N, and then order line N+1 would be picked first.

If two orders request the same tray, the model will stop. For example, when order A and B request the tray, operators A and B would walk to the VLM to pick up their item. Then, since order A requested the tray first, operator A would pick up its item. However, operator B would also pick its item simultaneously. When a VLM receives instructions, it displays a tray and waits until the operator has picked the item, and then the process continues after the operator has picked the item. The same happens when orders A and B request the same tray, giving two instructions to the VLM.

Further, operators A and B pick from the tray simultaneously, but only from order A's instructions to the VLM. When the instructions from order B arrives at the VLM, the *waiting for operator* loops infinitely because the operator has already picked the items. However, the VLM does not know that the item is picked. The event that the same tray is requested multiple times simultaneously occasionally, mainly when CBS is applied. Therefore, the *acquire bay* activity ensures that the model still runs, even when a given tray is requested multiple times simultaneously.

7.4.2 Exported data and data processing

Some data should have been exported from the model, and some analyses should have been executed differently. In Section 3.1.1, it is mentioned that when the replenishment policy and level changes, the frequency and order size of the replenishment orders might be affected. In Section 6.1.2, it is shown that both the policy and the level affect the number of orders. However, the order size is not mentioned. This is because the replenishment order size was not recorded during the simulation runs, so the data was not analysed. This data would have been used to verify the size of each replenishment order. The need for this data was not noticed until after the scenarios were run and towards the end of the thesis timeline. When it was noticed, there was not enough time to rerun all scenarios to get the required data.

Further, most of the data processing was executed in Excel because of the author's previous experience with the software. Here, most of the data has been reduced from thousands of data points to only a few points which shows the averages between each scenario. When using averages, an important aspect that needs to be mentioned is the flaw of averages. An average will often not give the whole picture, with or without the standard deviation, because much of the information is lost when calculating the average. The fluctuations between each scenario or each order will not be shown when only one number is given. Also, when using averages, extreme points will draw the average towards a higher or lower value, which might not give a satisfactory value. Although the average has multiple flaws, it can be used to show a clear picture if the data set is large enough.

Further, the standard deviation might give a clearer picture when combined with the average. A high standard deviation compared to the average value means higher uncertainty within the average value, and the data points are spread out. This is the fault within the data analysed in this thesis, the average seems fine, but compared to the standard deviation, the averages do not show a clear picture of the different scenarios.

The data have variation between each scenario, and the standard deviation of each scenario is also high compared to the average. The average standard deviation between each scenario is used to calculate the standard deviations shown in Section 6.1. The formula is shown in Equation 7.1.

$$\text{Average Standard Deviation} = \sqrt{\frac{(s_1)^2 + (s_2)^2 + \ldots + (s_n)^2}{N}} \tag{7.1}$$

s = Standard deviation of value in scenario n

N = Number of scenarios

Lastly, all the data processing and analyses could have been executed in SPSS to give better and more results. Better results in the way that the flaw of averages could have been avoided by using other statistical tools, and these analyses could have given a more precise result. Further, analyses could have been carried out on the average picking time per item and the average picking time per order. All the raw data could have been processed in SPSS to show the differences between each scenario better. Due to the flaw of averages, the regression analyses were carried out to better understand the relationship between the warehouse policies and the values investigated.

Chapter 8

Conclusion

This study aimed to investigate multiple policies for warehousing functions and activities and to examine how these affect a system where multiple VLMs operate in sequence. First, the theoretical background of warehousing and its functions and activities are reviewed. Further, multiple replenishment, put-away, storage and order picking policies are listed. Then follows a literature review to map the current research of VLMs, where there is a clear research gap regarding replenishment and put-away and multiple VLMs operating in sequence. Further, a case study was conducted where a detailed explanation of LS HMN is shown. The case study and theoretical backgrounds were used to answer RQ1. The case study was then used as a description to build a DES model where the effect of changes in the applicable warehouse policy could be modelled. The model was used to give quantitative results needed to RQ2, 3 and 4.

RQ1 What are the applicable warehouse policies for a system with multiple VLMs working in sequence?

For a system with multiple VLMs working in sequence, there are different policies applicable for each warehouse function and activity. The main functions and activities analysed and discussed in this thesis are replenishment, put-away, storage and order picking. For each of these, there are multiple applicable policies and configurations. These are seen in Table 8.1.

Table 8.1: Applicable warehouse policies for a system with multiple VLMs operating in parallel

Warehouse functions and activities	Applicable policies	
Replenishment	Two-bin system	Individual tray Each VLM
	ROP FOI	Total stock Individual VLM
Put-away	Zone Zone-Batch	
Storage policy	Random Dedicated OOS	
	CBS	Within each VLM Between VLMs
Order picking	Zone Zone-batch	

RQ2 What is the impact on order picking time in a system with multiple VLMs when replenishment and put-away are executed in parallel with order picking?

When executed in parallel with order picking, the order picking time is not directly affected by replenishment and put-away. However, the stay time of an order is affected because more tasks should be executed, which is further affected by the sequencing of tasks. In a real system, the waiting time might increase if replenishment and put-away are executed in parallel with order picking. The number of tasks increases, and if multiple operators are used, the chance of waiting for other operators to finish is higher compared to when only one operator is used. Also, increasing the number of operators would increase the number of active VLMs, which would further increase the chances of waiting.

RQ3 What is the impact on the order picking time when changing between the applicable warehouse policies in a system where multiple VLMs operate in sequence?

Firstly, the replenishment policy does not affect the order picking time or wait during order picking, only the stay time for the orders. When ROP is used, the stay time of an order is the lowest. Further, the replenishment level affects both the order picking time and the average stay time of an order. If the replenishment level is set to total, the order picking time, wait during order picking and stay time are lower than if the level is set to individual VLMs. Further, the put-away policy barely affects the order picking time and the replenishment execution time; both are executed a fraction quicker when batch is used compared to discrete. However, when the batch is used, the stay time for an order is reduced by 21%, and it is reduced by 51% for a replenishment order. The decrease occurs because the operators walk less, and the pick-up time for a replenishment order is the same, no matter the number of items.

Further, the change in storage policy from random to CBS decreases the order picking time and replenishment execution time. As a result, the average stay time for both orders and replenishment orders is also reduced. For order picking, there is not a big difference between discrete and batch picking, according to the results. When batch picking is used, the order is picked a minuscule amount quicker, and the waiting time is increased by a second, but the stay time is more than doubled compared to discrete order picking.

The combination of ROP at the total level, with batch put-away, CBS storage and batch order picking gives the quickest order picking time.

RQ4 What is the impact on the order picking time when the number of operators and active VLMs are altered?

Increasing the number of operators does not affect the order picking time or the waiting time during order picking in the model. However, increasing the active VLMs parameter will drastically reduce the order picking time and the waiting time during order picking. The time reduction of setting the parameter to five instead of one reduces the order picking and waiting time by 42% and 76%, respectively. Further, the replenishment and put-away execution time is not affected by changes in the active VLMs parameter. However, when the amount of operators increases, the execution and waiting time increase. Lastly, the stay time of both orders and replenishment orders is significantly reduced when two operators are used instead of one. The stay time is lowered when more operators are used, but the effects on the stay time decrease when the number of operators increases.

The thesis has contributed to the research gaps highlighted. The thesis extends the research on how changes in warehouse policies affect how multiple VLMs operate in sequence. Also, the thesis fills the research gap of how replenishment and put-away affect the order picking time when executed in parallel with order picking in a system with multiple VLMs. Further, the thesis can give guidelines for which warehouse policies a company could apply to reduce the order picking time, especially if the company uses multiple VLMs.

8.1 Further work

Regarding the simulation model, multiple steps can be taken to give more realistic and accurate results. Firstly, enabling multiple picks from the same tray without storing and retrieving the tray between each pick would make the model more realistic. Also, fixing the scheduling and deleting of order line logic would further improve the model's accuracy.

This thesis has mentioned multiple aspects that have yet to be examined. Firstly, the effects on replenishment order size should be further examined when the replenishment level is changed. Intuitively, when the model has the same output during a period, but the frequency of orders changes, the order size should also change. This thesis has mentioned but not examined the change in order sizes. Also, the OOS storage policy could further reduce the order picking time if multiple items of the same order are stored on the same tray. The effects of this policy when multiple VLMs operating in sequence should be researched.

The simulation model used in this thesis only recorded order picking time. It would have been interesting to see how these policy changes affect the time usage of the VLMs. For example, how much time is the VLMs idle or waiting for the operator. Further, the idle time of the operators could also have been examined, especially if the system is run at peak capacity. Lastly, it would have been interesting to examine how changes in the warehouse policies and parameters affect the system when it is run at peak capacity.

Bibliography

Battini, Daria et al. (2015). 'Order picking system design: the storage assignment and travel distance estimation (SA&TDE) joint method'. In: *International Journal of Production Research* 53.4, pp. 1077–1093. DOI: 10.1080/00207543.2014.944282.

Blikås, Mikael H., Sigmund Pedersen and Lars I. Vassbø (2021). *Application of Discrete Event Simulation to Optimize the Efficiency of a Miniload Automated Storage and Retrieval System at ASKO Sentrallager Kjøl AS*. Bachelor's Thesis.

Calzavara, M., F. Sgarbossa and A. Persona (2019). 'Vertical Lift Modules for small items order picking: an economic evaluation'. In: *International Journal of Production Economics* 210, pp. 199–210. DOI: 10.1016/j.ijpe.2019.01.012.

Daria, B. et al. (2015). 'Dual-tray vertical lift module for order picking: A performance and storage assignment preliminary study'. In: Industrial Systems Engineering, pp. 85–90. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84949567184%5C&partnerID=40%5C&md5=25890176e8524d8f38e21e384b66fc64.

de Koster, René, Tho Le-Duc and Kees Jan Roodbergen (2007). 'Design and control of warehouse order picking: A literature review'. In: *European Journal of Operational Research* 182.2, pp. 481–501. ISSN: 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2006.07.009.

Dukic, G., T. Opetuk and T. Lerher (2015). 'A throughput model for a dual-tray Vertical Lift Module with a human order-picker'. In: *International Journal of Production Economics* 170, pp. 874–881. DOI: 10.1016/j.ijpe.2015.04.009.

FlexSim® (2023a). *3D Simulation Modeling and Analysis Software*. URL: https://www.flexsim.com (visited on 4th May 2023).

— (2023b). *Math*. URL: https://docs.flexsim.com/en/21.2/Reference/CodingInFlexSim/FlexScriptAPIReference/Data/Math.html#Method-floor (visited on 19th May 2023).

— (2023c). *TASK 1.3 - BUILD A PROCESS FLOW MODEL*. URL: https://docs.flexsim.com/en/19.1/Tutorials/FlexSimBasics/1-3BuildProcessFlow/1-3BuildProcessFlow.html (visited on 4th May 2023).

— (2023d). *The experimenter*. URL: https://docs.flexsim.com/en/22.0/Reference/Tools/Experimenter/Experimenter.html (visited on 31st May 2023).

Frazelle, Edward H. (2016). *World-Class Warehousing and Material Handling. 2nd ed.* McGraw-Hill Education. ISBN: 9780071842822. URL: https://www.accessengineeringlibrary.com/content/book/9780071842822.

Hackman, John J. Bartholdi III & Steven T. (2019). *Warehouse & Distribution science.* URL: https://www.warehouse-science.com.

Kay, Michael G. (2015). *Warehousing.* URL: https://mgkay.github.io/Warehousing.pdf.

Kleijnen, Jack P.C. (1995). 'Verification and validation of simulation models'. In: *European Journal of Operational Research* 82.1, pp. 145–162. ISSN: 0377-2217. DOI: https://doi.org/10.1016/0377-2217(94)00016-6.

Lam, C.H.Y. et al. (2009). 'Development of an OLAP Based Fuzzy Logic System for Supporting Put Away Decision'. In: *International Journal of Engineering Business Management* 1, p. 13. DOI: 10.5772/6779.

Law, Averill M. (2015). *Simulation Modeling and Analysis Fifth Edition.* McGraw-Hill Education. ISBN: 9780073401324.

Mantel, R.J., P.C. Schuur and S.S. Heragu (2007). 'Order oriented slotting: a new assignment strategy for warehouses'. In: *European Journal of Industrial Engineering* 1.3, pp. 301–316. DOI: 10.1504/EJIE.2007.014689.

Meller, R.D. and J.F. Klote (2004). 'A throughput model for carousel/VLM pods'. In: *IIE Transactions (Institute of Industrial Engineers)* 36.8, pp. 725–741. DOI: 10.1080/07408170490458472.

MHI (2023a). *Material Handling.* URL: https://www.mhi.org/fundamentals/material-handling (visited on 16th Mar. 2023).

— (2023b). *Vertical Lift Modules.* URL: https://www.mhi.org/solutions-community/solutions-guide/vertical-lift (visited on 16th Mar. 2023).

Nicolas, L., F. Yannick and H. Ramzi (2016). 'Optimization of order batching in a picking system with a Vertical Lift Module: 6th International Conference on Information Systems, Logistics and Supply Chain (ILS International Conference)'. In: URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84985955413%5C&partnerID=40%5C&md5=6681d1d37d4775a822820131e50eb71d.

— (2018). 'Order batching in an automated warehouse with several vertical lift modules: Optimization and experiments with real data'. In: *European Journal of Operational Research* 267.3, pp. 958–976. DOI: 10.1016/j.ejor.2017.12.037.

Norge, Helse-Midt (2022). *Om oss.* URL: https://helse-midt.no/om-oss (visited on 28th Nov. 2022).

Oppen, Marit, Bjørn Erik Mørk and Eirik Haus (2020). *Kvantitative og kvalitative metoder i merkantile fag en introduksjon.* Cappelen Damm AS. ISBN: 9788202410803.

Ridley, Diana (2012). *The Literature Review A Step-by-Step Guide for Students.* Sage. ISBN: 9781446201428.

Robinson, Stewart (2004). *Simulation: The Practice of Model Development and Use.* John Wiley & Sons, Ltd. ISBN: 9780470847725.

Sargent, Robert G. (2020). 'Verification and Validation of Simulation Models: An Advanced Tutorial'. In: vol. 2020-December. Cited by: 16, pp. 16–29. DOI: 10.1109/WSC48552.2020.9384052.

Sgarbossa, F., M. Calzavara and A. Persona (2017). 'Economic and performance analysis of dual-bay Vertical Lift Modules (VLMs)'. In: URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85084160441%5C&partnerID=40%5C&md5=003ebdd52eb9b2b013bc7d1971262bb8.

— (2019). 'Throughput models for a dual-bay VLM order picking system under different configurations'. In: *Industrial Management and Data Systems* 119.6, pp. 1268–1288. DOI: 10.1108/IMDS-11-2018-0518.

Stevenson, William J. (2015). *Operations management, Twelfth Edition*. Presidio Press. ISBN: 9780078024108.

Tompkins, JA (2010). *Facilities Planning*. John Wiley Sons Inc. ISBN: 9780470444047.

Tranfield, David, David Denyer and Palminder Smart (2003). 'Towards a Methodology for Developing Evidence-Informed Management Knowledge by Means of Systematic Review'. In: *British Journal of Management* 14.3, pp. 207–222. DOI: https://doi.org/10.1111/1467-8551.00375.

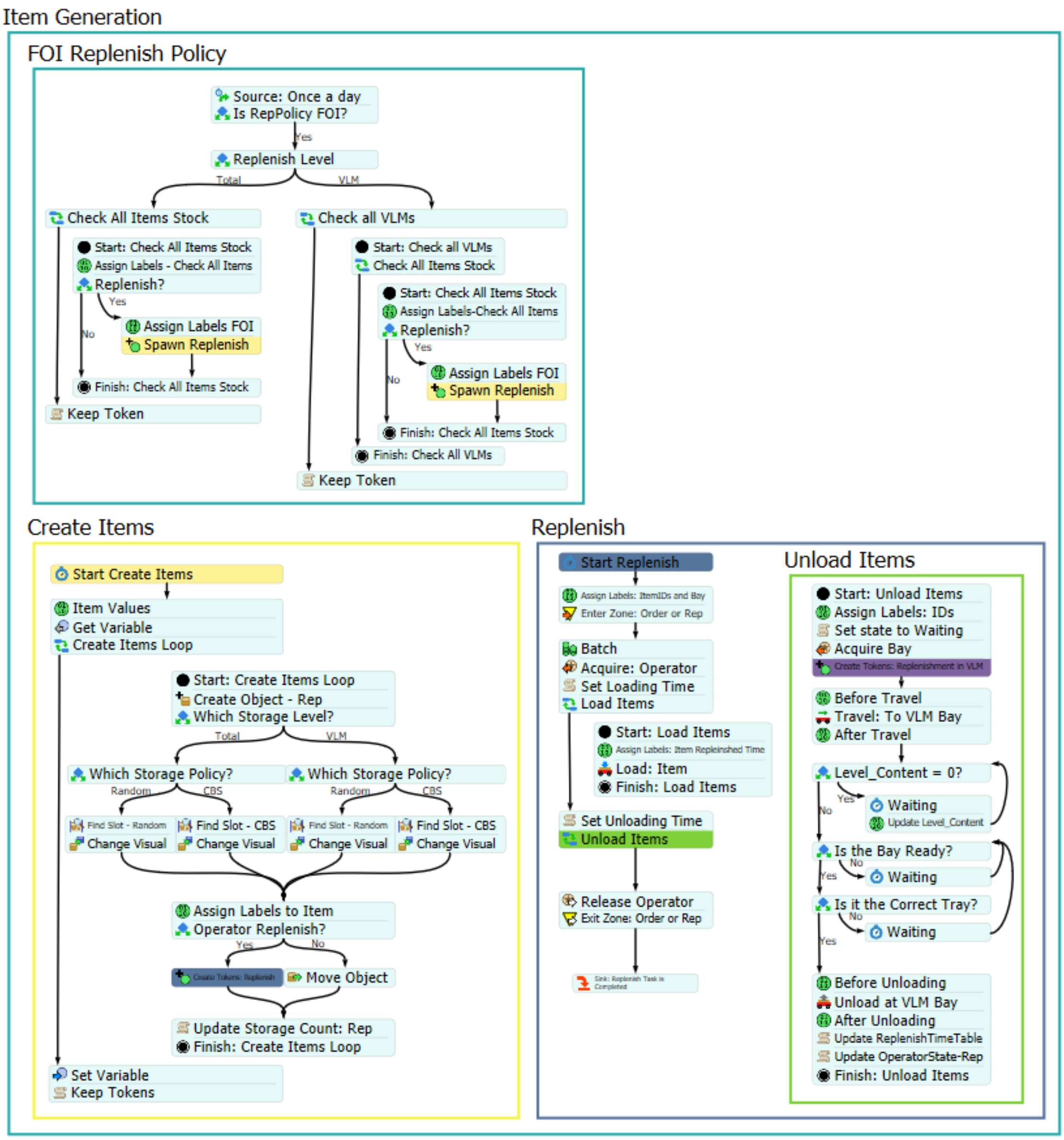
Vanhauwermeiren, P., M. Juwet and E. Demeester (2021). 'Pick capacity model for cutting machine tools stored in a next generation Vertical Storage Machine'. In: vol. 55. C. cited By 0, pp. 479–486. DOI: 10.1016/j.promfg.2021.10.065.

Voss, Chris, Nikos Tsikriktsis and Mark Frohlich (2002). 'Case research in operations management'. In: *International journal of operations & production management* 22.2, pp. 195–219. ISSN: 0144-3577.

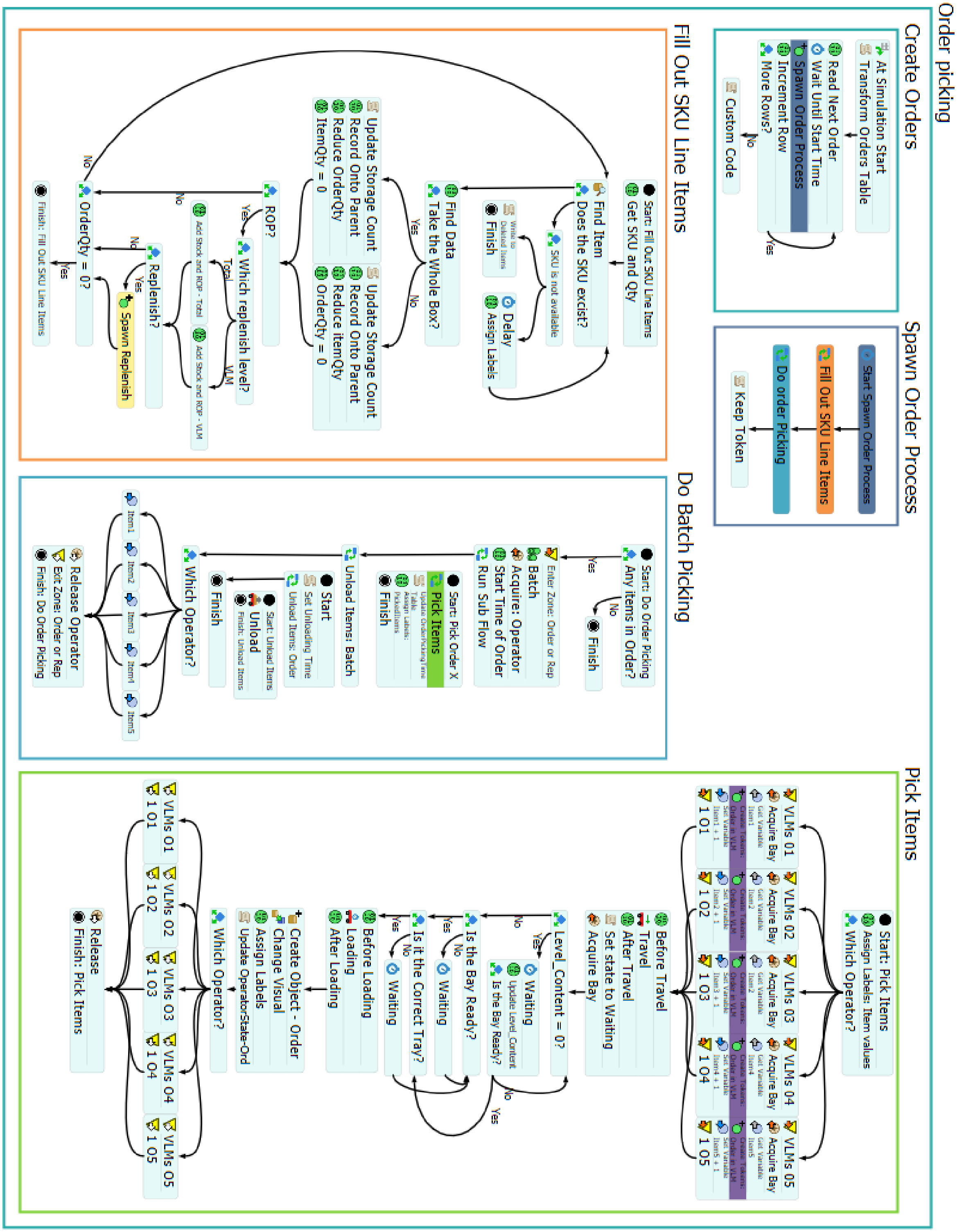
Walitsarangkul, A. and O. Kittithreerapronchai (2020). 'Storage Allocation for Stocked and Buffer Baskets'. In: vol. 893. 1. DOI: 10.1088/1757-899X/893/1/012009.

Appendices

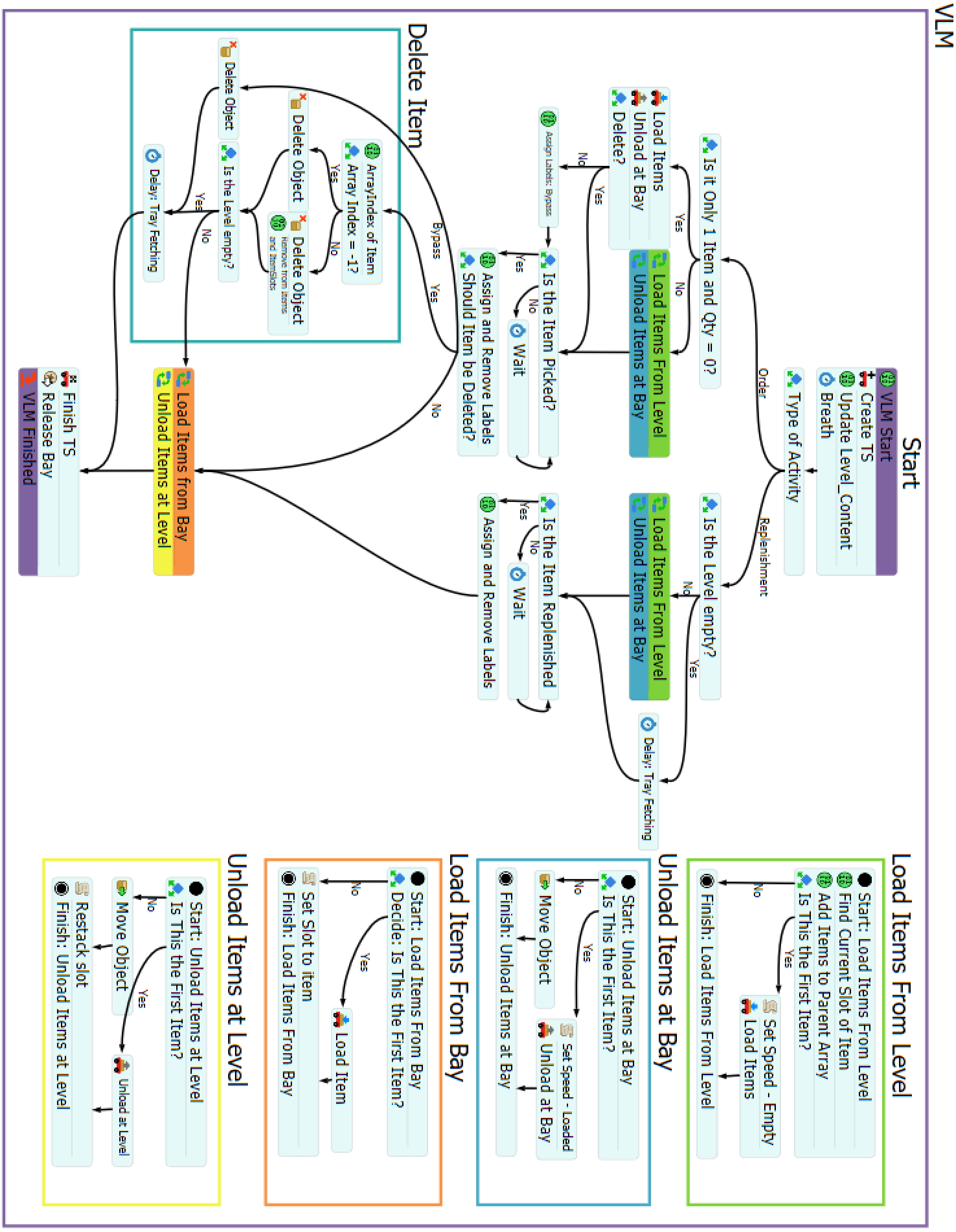
A Process Flow



Item generation process flow



Order releasing and Order picking process flow



VLM process flow

B Results numbers

Data used for the results section. Averages from each scenario and all values are in seconds.

Start of table								
Scenario	**Average order picking time**	**Average time for executing one replenishment order**	**Average wait during order picking**	**Average wait during replenishment**	**Average stay time for an order**	**Average stay time for a replenishment order**	**Average total time spent during order picking**	**Average total time spent during replenishment**
1	214	-	65	-	577	-	306 994	-
2	210	-	63	-	3581	-	300 990	-
3	218	319	67	245	4864	14665	311 934	246 615
4	217	326	71	251	7263	15606	311 229	250 290
5	216	275	66	221	2761	5308	308 309	208 971
6	214	286	67	232	5602	5804	307 070	223 230
7	207	-	57	-	553	-	296 348	-
8	203	-	55	-	3508	-	290 848	-
9	209	297	59	221	4558	14359	298 267	223 710
10	207	301	60	225	6678	14760	296 385	226 551
11	206	266	56	211	2677	5261	294 710	201 148
12	205	294	59	238	5500	5673	294 271	220 397
13	213	-	64	-	569	-	305 455	-
14	209	-	64	-	3557	-	299 985	-
15	214	168	65	87	2683	10662	306 262	128 190
16	213	169	67	87	5073	10618	304 446	128 139
17	214	179	66	110	1710	3540	307 140	136 598
18	210	179	65	109	4145	3479	301 050	134 817
19	205	-	56	-	535	-	293 924	-
20	201	-	55	-	3488	-	288 208	-
21	206	153	57	71	2387	10135	295 553	116 174
22	204	154	59	72	4768	10015	292 372	115 937
23	207	163	58	94	1496	3326	296 743	123 786
24	202	162	56	94	3950	3210	290 399	123 779
25	216	-	65	-	582	-	310 539	-
26	213	-	65	-	3552	-	306 861	-
27	213	112	66	72	1258	1545	307 296	146 840
28	213	96	67	55	3906	1336	306 309	127 002
29	216	101	66	68	950	3673	310 024	132 126
30	211	103	65	70	3734	3615	302 845	134 435
31	208	-	58	-	552	-	299 902	-
32	203	-	56	-	3440	-	292 975	-
33	206	105	58	63	1099	1383	295 977	136 163
34	204	90	58	49	3792	1192	293 449	115 123
35	207	92	58	59	840	3512	297 064	121 188
36	204	94	58	61	3733	3579	293 115	120 532
37	213	-	64	-	571	-	306 739	-
38	213	-	64	-	3546	-	305 806	-
39	215	247	67	131	1362	6607	308 771	125 729
40	213	252	67	135	4131	6610	306 493	133 194
41	213	262	64	162	964	3583	305 505	136 251

Continuation of Table 2								
Scenario	**Average order picking time**	**Average time for executing one replenishment order**	**Average wait during order picking**	**Average wait during replenishment**	**Average stay time for an order**	**Average stay time for a replenishment order**	**Average total time spent during order picking**	**Average total time spent during replenishment**
42	211	268	65	169	3814	3539	302 490	138 049
43	206	-	55	-	558	-	296 221	-
44	204	-	56	-	3448	-	247 361	-
45	204	222	57	107	1137	5966	294 349	114 728
46	204	222	59	105	3943	5564	293 129	115 455
47	206	243	57	140	852	3444	295 991	125 261
48	203	237	57	138	3653	3323	290 988	119 841
End of Table								

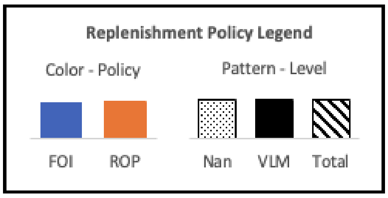
Standard deviations for each scenario. Values are in seconds.

Start of table						
Scenario	**Order picking time**	**Time for executing one replenishment order**	**Wait during order picking**	**Wait during replenishment**	**Stay time for an order**	**Stay time for a replenishment order**
1	161	-	47	-	453	-
2	159	-	46	-	1 318	-
3	165	174	50	138	5 003	9 502
4	165	178	53	141	3 266	10 437
5	164	155	49	131	3 012	1 848
6	162	162	49	138	2 422	1 994
7	155	-	43	-	432	-
8	152	-	40	-	1 294	-
9	157	158	44	123	4 878	9 173
10	157	161	45	125	3 001	9 564
11	154	150	41	125	3 268	1 779
12	156	166	44	140	2 342	1 923
13	161	-	48	-	434	-
14	158	-	46	-	1 316	-
15	163	97	49	53	2 892	7 337
16	161	96	50	52	2 065	7 318
17	162	103	49	66	1 847	1 192
18	158	102	47	66	1 585	1 175
19	154	-	41	-	418	-
20	150	-	39	-	1 285	-
21	154	84	42	41	2 638	6 976
22	153	86	43	43	1 913	6 872
23	156	91	43	55	1 648	1 152
24	152	91	42	56	1 498	1 082

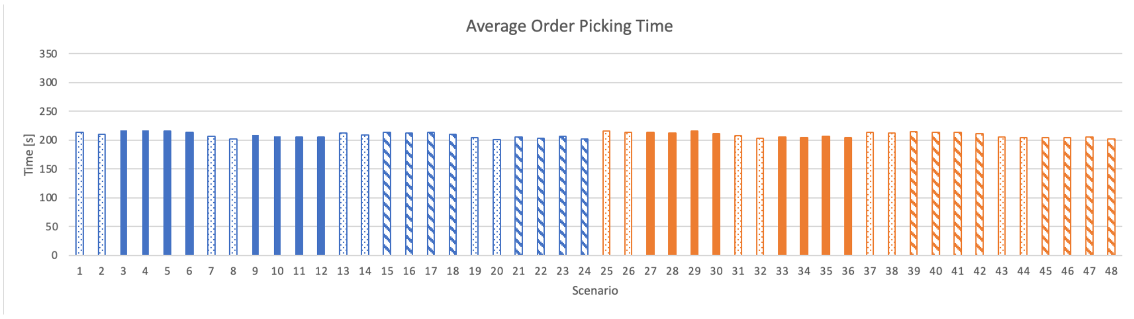
Continuation of Table 3						
Scenario	**Order picking time**	**Time for executing one replenishment order**	**Wait during order picking**	**Wait during replenishment**	**Stay time for an order**	**Stay time for a replenishment order**
25	163	-	49	-	455	-
26	161	-	49	-	1 304	-
27	162	69	49	51	1 033	1 176
28	161	58	48	40	1 431	1 014
29	163	61	49	47	809	832
30	158	64	46	50	1 358	848
31	158	-	45	-	424	-
32	153	-	41	-	1 267	-
33	155	60	43	43	876	1 029
34	154	50	41	33	1 391	982
35	156	54	43	40	669	798
36	154	54	42	39	1 366	845
37	161	-	47	-	446	-
38	160	-	47	-	1 302	-
39	163	174	50	92	1 225	7 659
40	161	182	48	98	1 549	7 693
41	160	195	47	125	843	1 179
42	159	189	47	121	1 400	1 177
43	155	-	42	-	434	-
44	149	-	39	-	1 179	-
45	154	147	42	66	1 020	7 097
46	153	149	42	68	1 469	6 334
47	155	165	42	93	724	1 120
48	153	155	41	88	1 349	1 090
End of Table						

C Results graphs

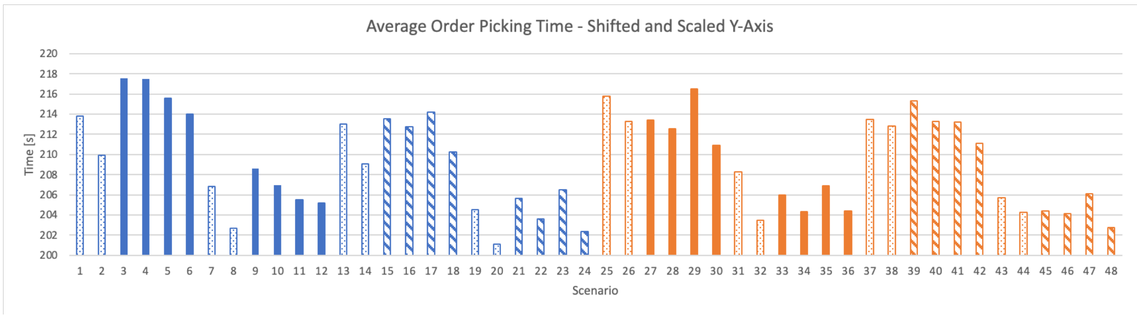
Change of replenishment policy



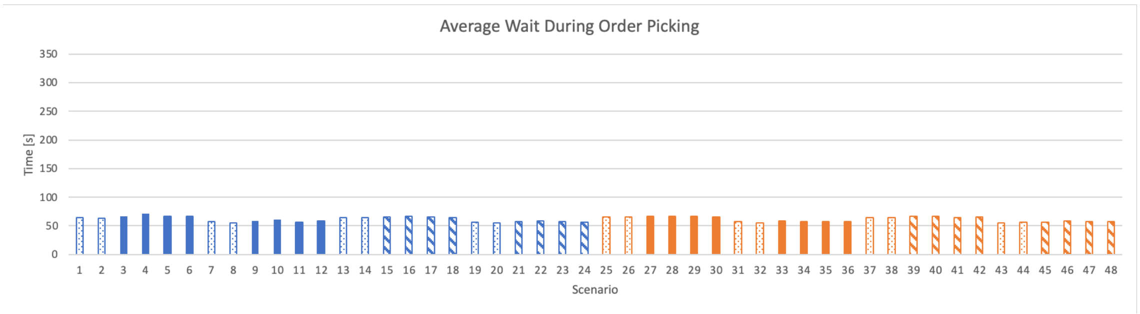
Change of replenishment policy legend



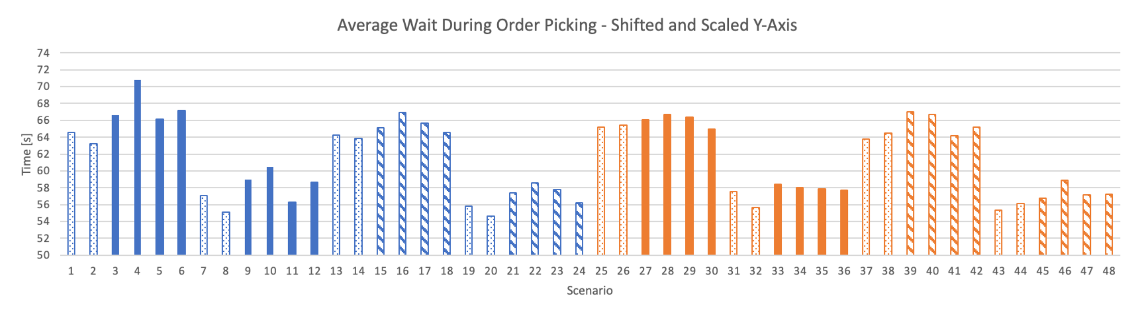
Average time for picking one order



Average time for picking one order - Shifted and scaled Y-axis

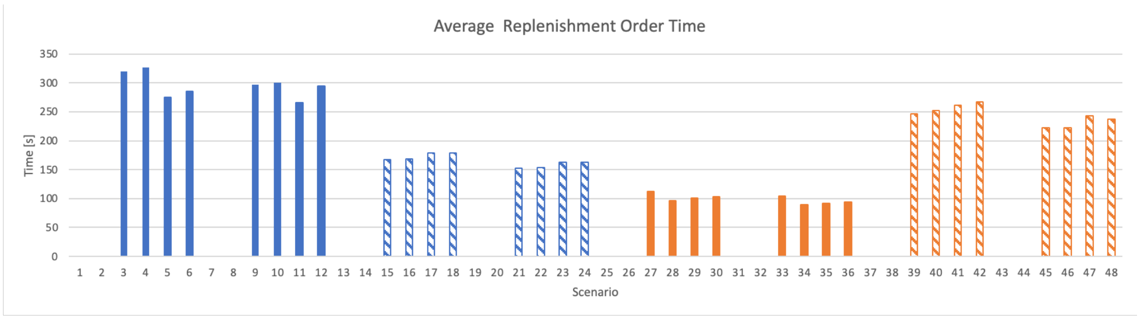


Average wait during order picking

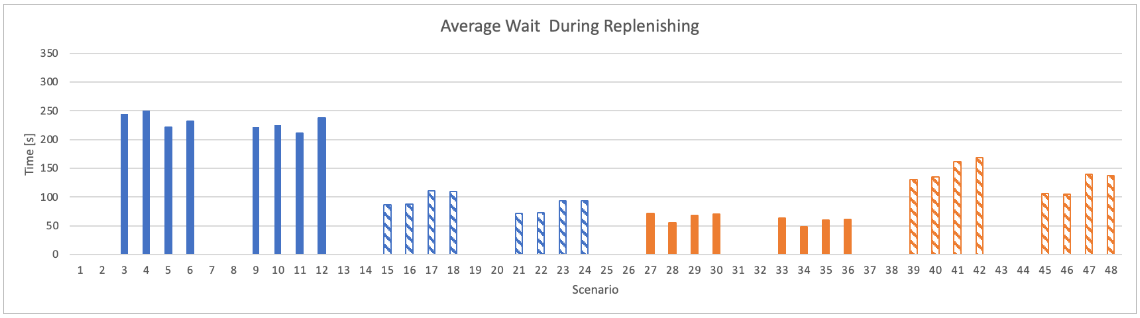


Average wait during order picking - Shifted and scaled Y-axis

Change of replenishment policy - Average time to pick one order and average wait during order picking.

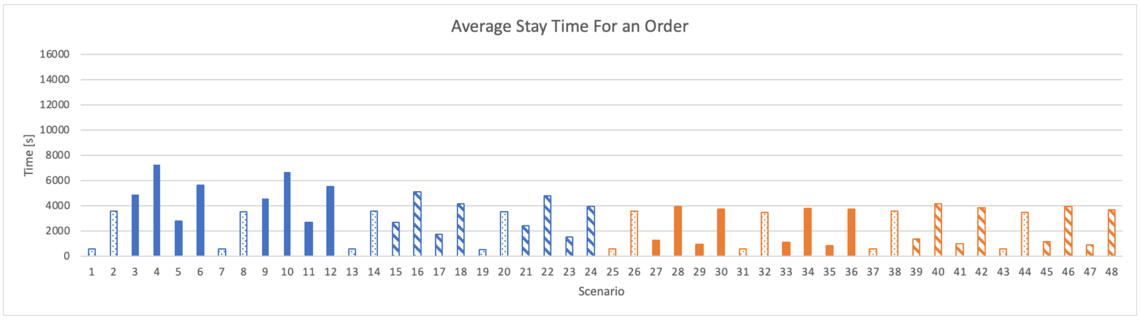


Average time for executing one replenishment order

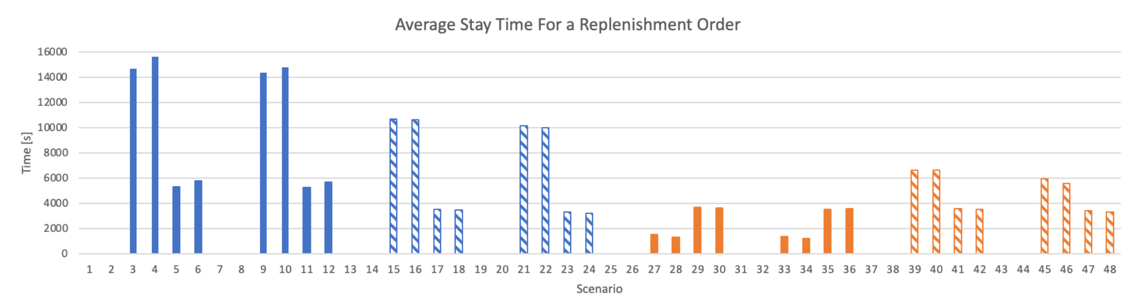


Average wait during replenishment

Change of replenishment policy - Average time to execute one replenishment order and average wait during replenishment.



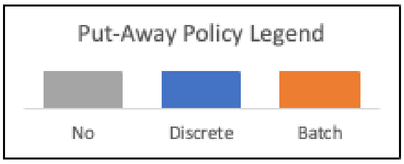
Average stay time for an order



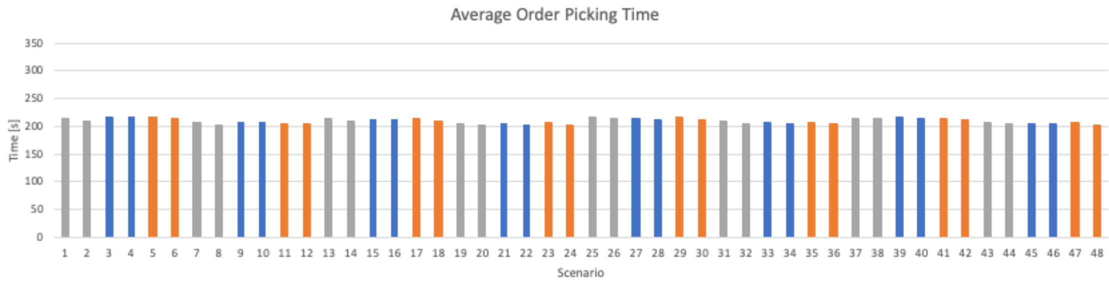
Average stay time for a replenishment order

Change of replenishment policy - Average stay time for an order or a replenishment order.

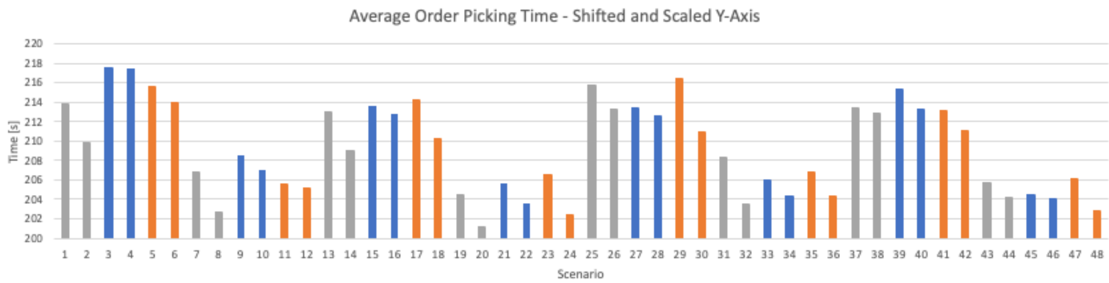
Change of put-away policy



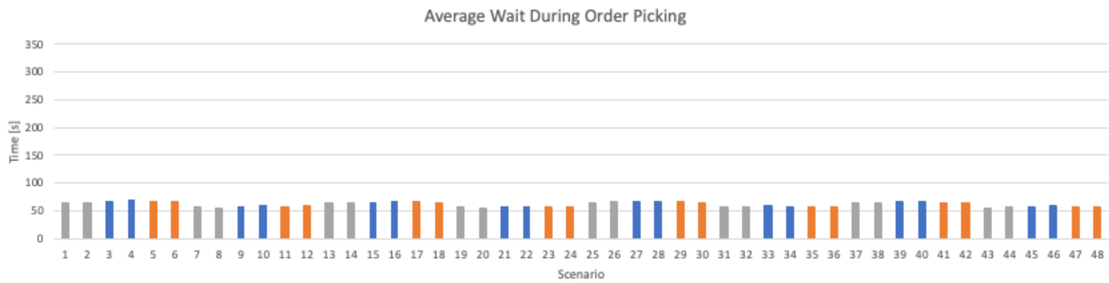
Change of put-away policy legend



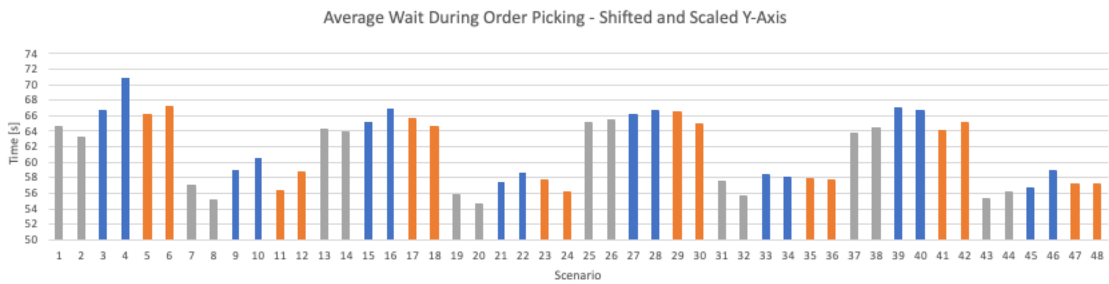
Average time for picking one order



Average time for picking one order - Shifted and scaled Y-axis

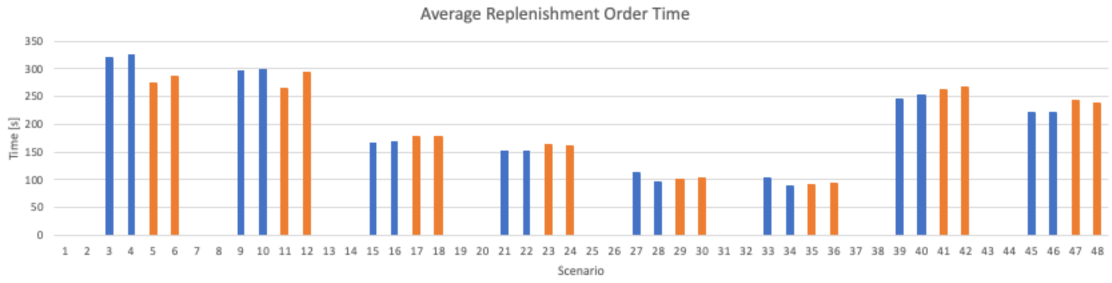


Average wait during order picking

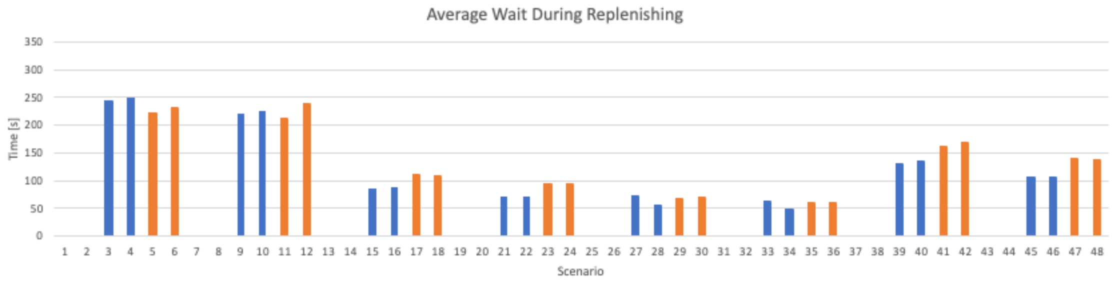


Average wait during order picking - Shifted and scaled Y-axis

Change of put-away policy - Average time to pick one order and average wait during order picking.



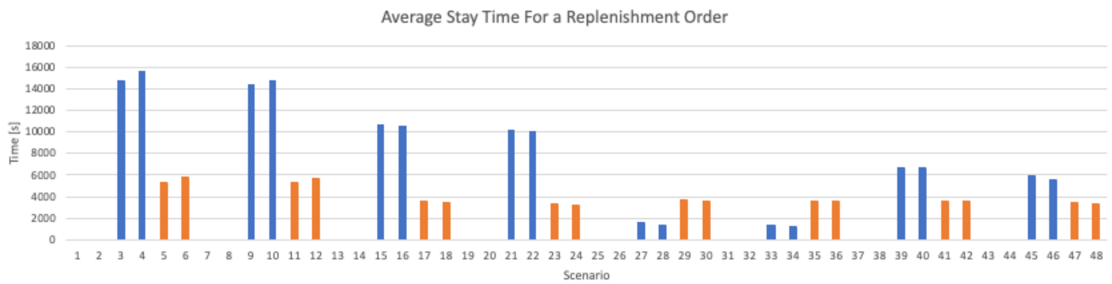
Average time for executing one replenishment order



Average wait during replenishment

Change of put-away policy - Average time to execute one replenishment order and average wait during replenishment.

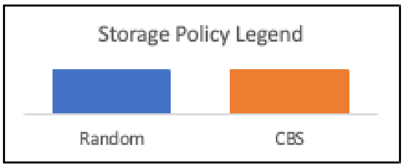
Average stay time for an order



Average stay time for a replenishment order

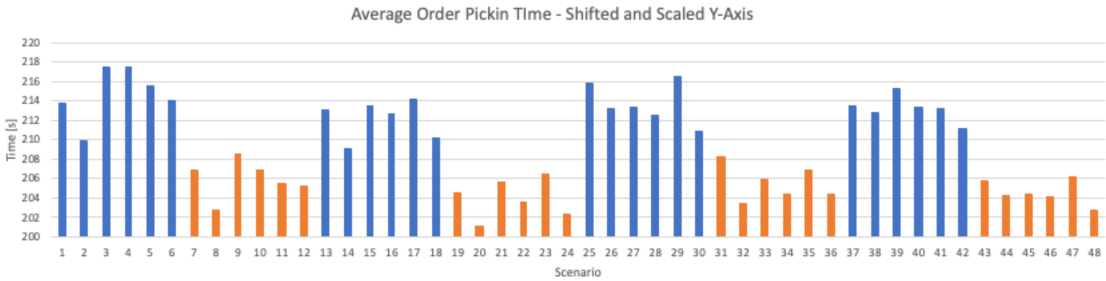
Change of put-away policy - Average stay time for an order or a replenishment order.

Change of storage policy

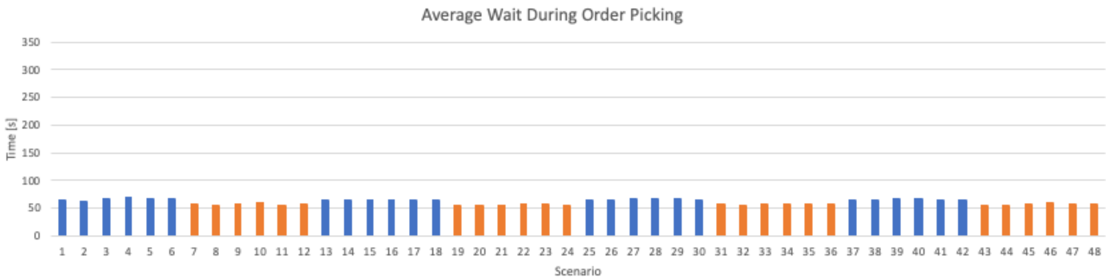


Change of storage policy legend

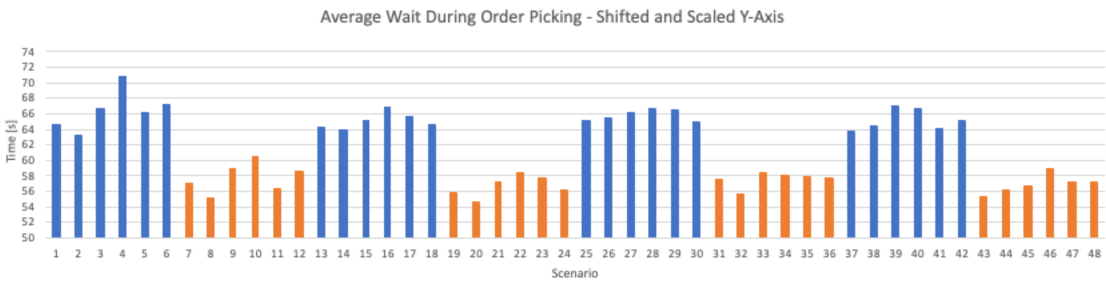
Average time for picking one order



Average time for picking one order - Shifted and scaled Y-axis

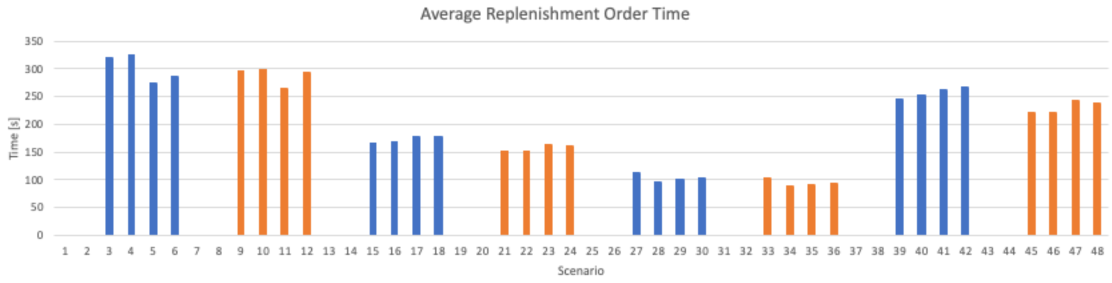


Average wait during order picking

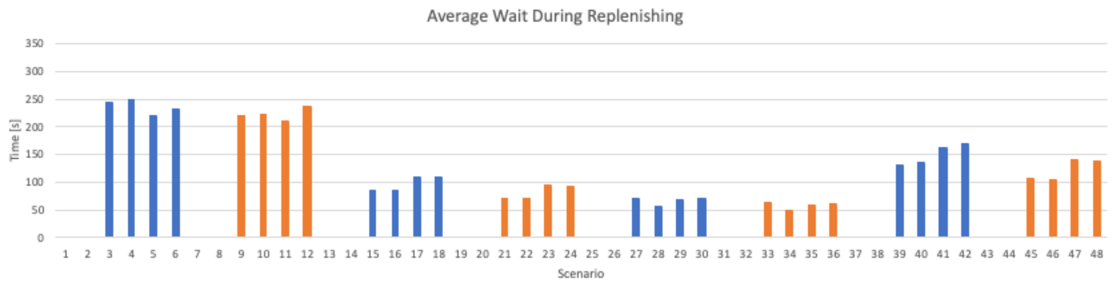


Average wait during order picking - Shifted and scaled Y-axis

Change of storage policy - Average time to pick one order and average wait during order picking.

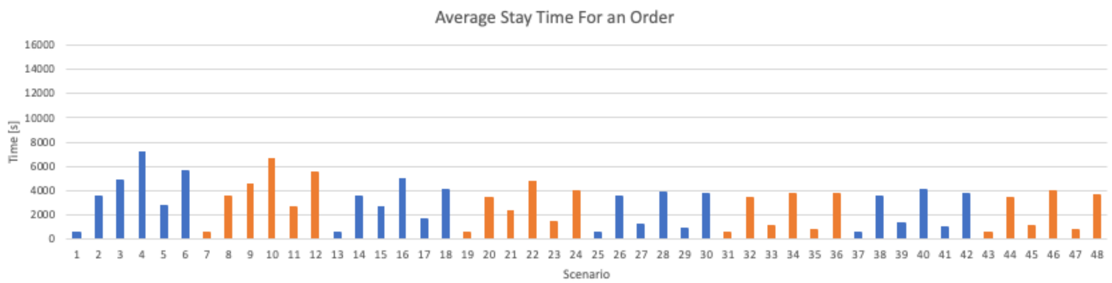


Average time for executing one replenishment order

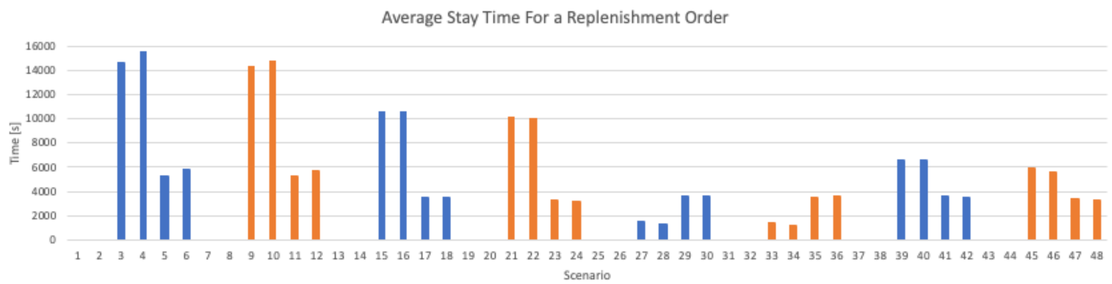


Average wait during replenishment

Change of storage policy - Average time to execute one replenishment order and average wait during replenishment.



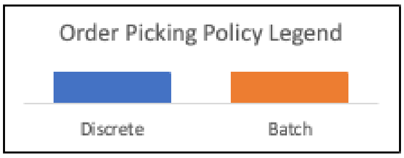
Average stay time for an order



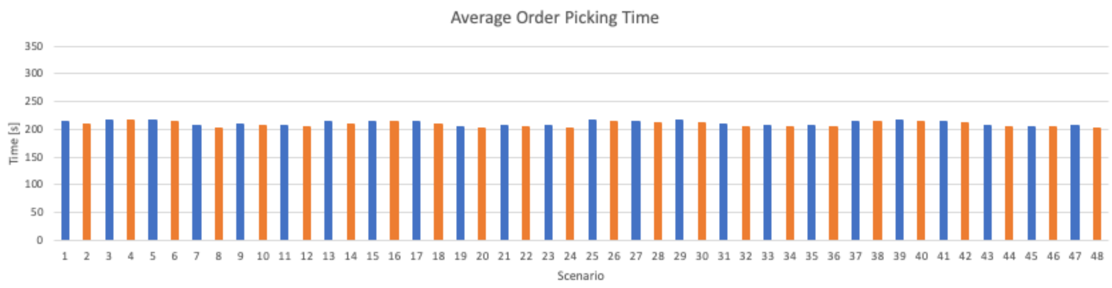
Average stay time for a replenishment order

Change of storage policy - Average stay time for an order or a replenishment order.

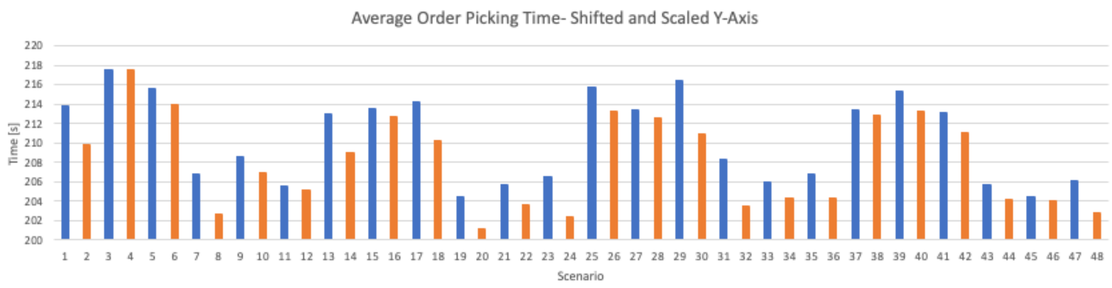
Change of order picking policy



Change of order picking policy legend

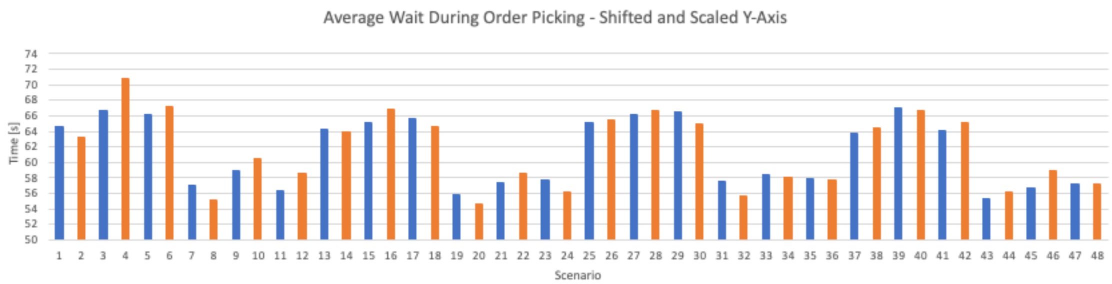


Average time for picking one order



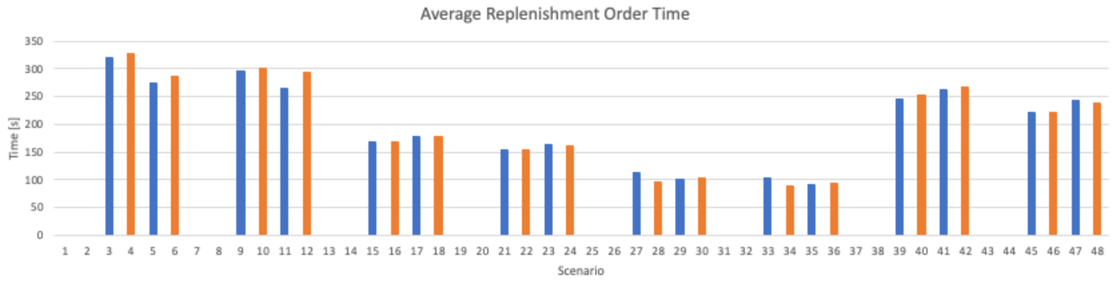
Average time for picking one order - Shifted and scaled Y-axis

Average wait during order picking

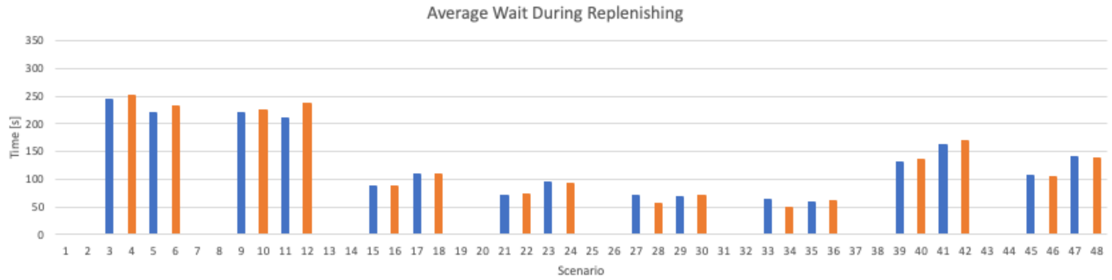


Average wait during order picking - Shifted and scaled Y-axis

Change of order picking policy - Average time to pick one order and average wait during order picking.

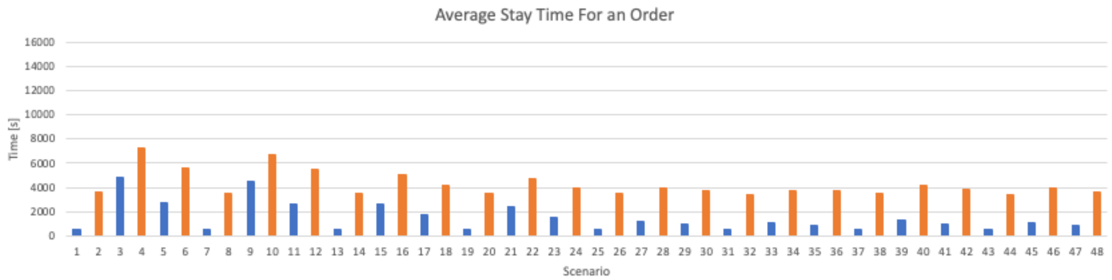


Average time for executing one replenishment order

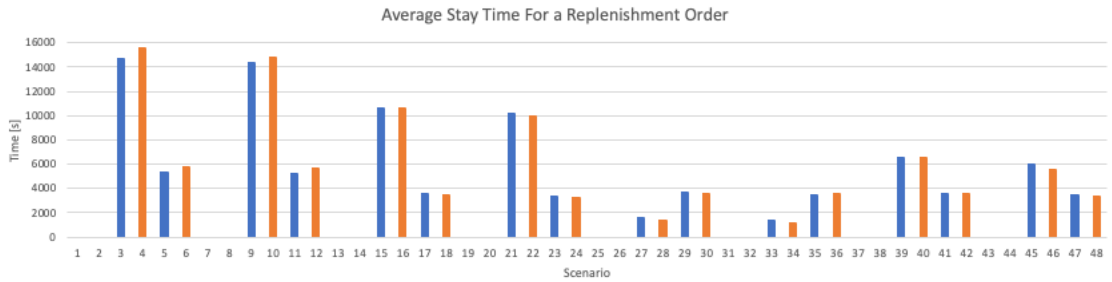


Average wait during replenishment

Change of order picking policy - Average time to execute one replenishment order and average wait during replenishment.



Average stay time for an order



Average stay time for a replenishment order

Change of order picking policy - Average stay time for an order or a replenishment order.

D Full regression analyses

Regressions - With and without replenishment

Order picking time

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Watson
1	.192[a]	.037	-.111	5.15263755	.037	.249	2	13	.783	.991

a. Predictors: (Constant), Put-Away, Replenish

b. Dependent Variable: Average order picking time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	13.228	2	6.614	.249	.783[b]
	Residual	345.146	13	26.550		
	Total	358.374	15			

a. Dependent Variable: Average order picking time

b. Predictors: (Constant), Put-Away, Replenish

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	95,0% Confidence Interval for B Upper Bound	Collinearity Statistics Tolerance	Collinearity Statistics VIF
1	(Constant)	209.659	2.231		93.969	<.001	204.839	214.479		
	Replenish	-.835	2.576	-.088	-.324	.751	-6.400	4.731	1.000	1.000
	Put-Away	1.616	2.576	.171	.627	.541	-3.950	7.181	1.000	1.000

a. Dependent Variable: Average order picking time

Wait during order picking

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Watson
1	.308[a]	.095	-.045	4.86031911	.095	.680	2	13	.524	.843

a. Predictors: (Constant), Put-Away, Replenish

b. Dependent Variable: Average wait during order picking

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	32.125	2	16.063	.680	.524[b]
	Residual	307.095	13	23.623		
	Total	339.220	15			

a. Dependent Variable: Average wait during order picking

b. Predictors: (Constant), Put-Away, Replenish

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	95,0% Confidence Interval for B Upper Bound	Collinearity Statistics Tolerance	Collinearity Statistics VIF
1	(Constant)	60.701	2.105		28.842	<.001	56.154	65.247		
	Replenish	-.470	2.430	-.051	-.194	.850	-5.720	4.780	1.000	1.000
	Put-Away	2.795	2.430	.303	1.150	.271	-2.455	8.045	1.000	1.000

a. Dependent Variable: Average wait during order picking

Total time during order picking

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.141[a]	.020	-.131	7509.73135	.020	.131	2	13	.878	.954

a. Predictors: (Constant), Put-Away, Replenish
b. Dependent Variable: Total time picking

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	14808762.6	2	7404381.300	.131	.878[b]
	Residual	733148844	13	56396064.9		
	Total	747957606	15			

a. Dependent Variable: Total time picking
b. Predictors: (Constant), Put-Away, Replenish

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Tolerance	VIF
1	(Constant)	300662.567	3251.809		92.460	<.001	293637.460	307687.673		
	Replenish	39.032	3754.866	.003	.010	.992	-8072.862	8150.926	1.000	1.000
	Put-Away	1923.712	3754.866	.141	.512	.617	-6188.182	10035.606	1.000	1.000

a. Dependent Variable: Total time picking

Average stay time for orders

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.647[a]	.419	.329	1772.72157	.419	4.685	2	13	.029	3.148

a. Predictors: (Constant), Put-Away, Replenish
b. Dependent Variable: Order stay time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	29442679.2	2	14721339.6	4.685	.029[b]
	Residual	40853043.2	13	3142541.782		
	Total	70295722.3	15			

a. Dependent Variable: Order stay time
b. Predictors: (Constant), Put-Away, Replenish

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Tolerance	VIF
1	(Constant)	2880.779	767.611		3.753	.002	1222.456	4539.101		
	Replenish	-1675.103	886.361	-.400	-1.890	.081	-3589.969	239.763	1.000	1.000
	Put-Away	2134.174	886.361	.509	2.408	.032	219.308	4049.040	1.000	1.000

a. Dependent Variable: Order stay time

Regressions - Changing warehouse policies

Order picking time

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Watson
1	.968[a]	.938	.926	1.30406153	.938	78.198	5	26	<.001	1.703

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish
b. Dependent Variable: Average order picking time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	664.912	5	132.982	78.198	<.001[b]
	Residual	44.215	26	1.701		
	Total	709.127	31			

a. Dependent Variable: Average order picking time
b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients B	Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	Upper Bound	Collinearity Statistics Tolerance	VIF
1	(Constant)	216.602	.565		383.587	<.001	215.441	217.763		
	Replenish	-.905	.461	-.096	-1.963	.060	-1.853	.042	1.000	1.000
	ReplenishLevel	-1.664	.461	-.177	-3.609	.001	-2.612	-.716	1.000	1.000
	Put-Away	-.885	.461	-.094	-1.919	.066	-1.832	.063	1.000	1.000
	Storage	-8.633	.461	-.917	-18.725	<.001	-9.581	-7.686	1.000	1.000
	OrderPicking	-2.051	.461	-.218	-4.449	<.001	-2.999	-1.104	1.000	1.000

a. Dependent Variable: Average order picking time

Wait during order picking

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Watson
1	.978[a]	.956	.948	1.01312590	.956	114.256	5	26	<.001	2.394

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish
b. Dependent Variable: Average wait during order picking

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	586.376	5	117.275	114.256	<.001[b]
	Residual	26.687	26	1.026		
	Total	613.063	31			

a. Dependent Variable: Average wait during order picking
b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients B	Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	Upper Bound	Collinearity Statistics Tolerance	VIF
1	(Constant)	67.319	.439		153.452	<.001	66.417	68.221		
	Replenish	-.515	.358	-.059	-1.437	.163	-1.251	.222	1.000	1.000
	ReplenishLevel	-1.006	.358	-.115	-2.808	.009	-1.742	-.269	1.000	1.000
	Put-Away	-1.257	.358	-.144	-3.510	.002	-1.994	-.521	1.000	1.000
	Storage	-8.366	.358	-.956	-23.357	<.001	-9.103	-7.630	1.000	1.000
	OrderPicking	.668	.358	.076	1.864	.074	-.069	1.404	1.000	1.000

a. Dependent Variable: Average wait during order picking

Total time during order picking

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Watson
1	.973[a]	.946	.936	1693.46449	.946	91.916	5	26	<.001	1.893

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish
b. Dependent Variable: Total time picking

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1317987279	5	263597456	91.916	<.001[b]
	Residual	74563371.4	26	2867821.976		
	Total	1392550651	31			

a. Dependent Variable: Total time picking
b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients B	Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	Upper Bound	Collinearity Statistics Tolerance	VIF
1	(Constant)	310488.199	733.292		423.417	<.001	308980.896	311995.501		
	Replenish	-771.546	598.730	-.058	-1.289	.209	-2002.254	459.161	1.000	1.000
	ReplenishLevel	-2285.784	598.730	-.173	-3.818	<.001	-3516.491	-1055.076	1.000	1.000
	Put-Away	-1531.715	598.730	-.116	-2.558	.017	-2762.422	-301.008	1.000	1.000
	Storage	-12150.578	598.730	-.921	-20.294	<.001	-13381.286	-10919.871	1.000	1.000
	OrderPicking	-2990.922	598.730	-.227	-4.995	<.001	-4221.629	-1760.215	1.000	1.000

a. Dependent Variable: Total time picking

Average stay time for orders

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics: R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Watson
1	.949[a]	.902	.883	610.282824	.902	47.609	5	26	<.001	.510

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish
b. Dependent Variable: Order stay time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	88658929.5	5	17731785.9	47.609	<.001[b]
	Residual	9683573.242	26	372445.125		
	Total	98342502.8	31			

a. Dependent Variable: Order stay time
b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients B	Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	Upper Bound	Collinearity Statistics Tolerance	VIF
1	(Constant)	3717.781	264.260		14.069	<.001	3174.587	4260.976		
	Replenish	-1684.311	215.768	-.480	-7.806	<.001	-2127.828	-1240.794	1.000	1.000
	ReplenishLevel	-821.725	215.768	-.234	-3.808	<.001	-1265.242	-378.209	1.000	1.000
	Put-Away	-782.576	215.768	-.223	-3.627	.001	-1226.092	-339.059	1.000	1.000
	Storage	-197.084	215.768	-.056	-.913	.369	-640.600	246.433	1.000	1.000
	OrderPicking	2630.392	215.768	.750	12.191	<.001	2186.876	3073.909	1.000	1.000

a. Dependent Variable: Order stay time

Replenishment time

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.404[a]	.164	.003	77.1570182	.164	1.017	5	26	.428	.303

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

b. Dependent Variable: Average replenishment order time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	30263.409	5	6052.682	1.017	.428[b]
	Residual	154783.342	26	5953.205		
	Total	185046.751	31			

a. Dependent Variable: Average replenishment order time

b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Tolerance	VIF
1	(Constant)	234.556	33.410		7.021	<.001	165.880	303.231		
	Replenish	-59.008	27.279	-.388	-2.163	.040	-115.081	-2.934	1.000	1.000
	ReplenishLevel	7.538	27.279	.050	.276	.784	-48.535	63.611	1.000	1.000
	Put-Away	-1.796	27.279	-.012	-.066	.948	-57.869	54.277	1.000	1.000
	Storage	-15.406	27.279	-.101	-.565	.577	-71.479	40.667	1.000	1.000
	OrderPicking	1.909	27.279	.013	.070	.945	-54.164	57.982	1.000	1.000

a. Dependent Variable: Average replenishment order time

Wait during replenishment

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.557[a]	.310	.177	60.3114388	.310	2.335	5	26	.071	.332

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

b. Dependent Variable: Average wait during replenishing

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	42460.971	5	8492.194	2.335	.071[b]
	Residual	94574.211	26	3637.470		
	Total	137035.182	31			

a. Dependent Variable: Average wait during replenishing

b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Tolerance	VIF
1	(Constant)	177.907	26.116		6.812	<.001	124.226	231.589		
	Replenish	-61.596	21.323	-.471	-2.889	.008	-105.427	-17.765	1.000	1.000
	ReplenishLevel	-33.124	21.323	-.253	-1.553	.132	-76.955	10.707	1.000	1.000
	Put-Away	12.549	21.323	.096	.589	.561	-31.282	56.380	1.000	1.000
	Storage	-15.992	21.323	-.122	-.750	.460	-59.823	27.838	1.000	1.000
	OrderPicking	1.760	21.323	.013	.083	.935	-42.071	45.590	1.000	1.000

a. Dependent Variable: Average wait during replenishing

Total time during replenishment

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.814[a]	.663	.599	28203.2456	.663	10.246	5	26	<.001	.459

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish
b. Dependent Variable: Total time replenishing

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	4.075E+10	5	8150068028	10.246	<.001[b]
	Residual	2.068E+10	26	795423062		
	Total	6.143E+10	31			

a. Dependent Variable: Total time replenishing
b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Tolerance	VIF
1	(Constant)	209121.955	12212.364		17.124	<.001	184019.082	234224.828		
	Replenish	-47900.928	9971.353	-.547	-4.804	<.001	-68397.338	-27404.518	1.000	1.000
	ReplenishLevel	-51149.625	9971.353	-.584	-5.130	<.001	-71646.035	-30653.216	1.000	1.000
	Put-Away	-3089.513	9971.353	-.035	-.310	.759	-23585.923	17406.897	1.000	1.000
	Storage	-13168.977	9971.353	-.150	-1.321	.198	-33665.387	7327.433	1.000	1.000
	OrderPicking	205.817	9971.353	.002	.021	.984	-20290.593	20702.226	1.000	1.000

a. Dependent Variable: Total time replenishing

Average stay time for replenishment order

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	.776[a]	.602	.525	2912.85450	.602	7.864	5	26	<.001	1.117

a. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish
b. Dependent Variable: Replenishment stay time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	333599021	5	66719804.3	7.864	<.001[b]
	Residual	220602755	26	8484721.359		
	Total	554201777	31			

a. Dependent Variable: Replenishment stay time
b. Predictors: (Constant), OrderPicking, Storage, Put-Away, ReplenishLevel, Replenish

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Tolerance	VIF
1	(Constant)	11005.573	1261.303		8.726	<.001	8412.928	13598.219		
	Replenish	-4871.655	1029.850	-.585	-4.730	<.001	-6988.541	-2754.769	1.000	1.000
	ReplenishLevel	-478.031	1029.850	-.057	-.464	.646	-2594.917	1638.855	1.000	1.000
	Put-Away	-4197.286	1029.850	-.504	-4.076	<.001	-6314.172	-2080.400	1.000	1.000
	Storage	-342.896	1029.850	-.041	-.333	.742	-2459.782	1773.990	1.000	1.000
	OrderPicking	59.594	1029.850	.007	.058	.954	-2057.292	2176.480	1.000	1.000

a. Dependent Variable: Replenishment stay time

NTNU

Norwegian University of Science and Technology

ST. OLAVS HOSPITAL

UNIVERSITETSSYKEHUSET I TRONDHEIM