Linköping University | Department of Management and Engineering
Master’s thesis, 30 ECTS | Logistics
202020 | LIU-IEI/LITH-EX-A--2020/001--SE

Product allocation for an automated order picking system in an e-commerce warehouse

– A data mining approach

Alexander Dahl (alex@dahl.dev)

Supervisor : Mike Malmgren
Examiner : Magnus Berglund

External supervisor : Maria Alriksson

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Abstract

Warehousing in the era of E-commerce has to be fast, correct and cheap. Warehouse automation is a measure E-commerce companies can take to get a more streamlined flow through their warehouse. Order picking is the most labor intensive task in a warehouse. By automating the order picking process companies can lower their costs and improve their response times.

This thesis studies the A-frame, an automated order picking system, at a large online pharmacy, Apotea AB. An A-frame has dispensing channels on its side and a conveyor belt that runs through the entire machine. Products for an order are ejected from the channels onto the conveyor belt and at the end of the machine they are dropped into a box. The box is then sealed, labeled and sent to the customer.

For the automatic flow to function correctly, all orders picked by the A-frame need to be complete orders. Complete orders are orders where there are no products missing. To maximize the throughput of the A-frame, an appropriate product allocation will be required. Due to the vast number of combinations, it is extremely difficult to identify an optimal product allocation.

This study has examined three different approaches to the product allocation problem for an A-frame. The first two methods are based on ranking the products depending on their quantities sold. The last method uses association rule learning, which is a machine learning technique for finding interesting patterns in a data set. Association rule learning was used to find which products were associated to each other. These associations were then placed in a graph structure and solved using a heuristic.

To evaluate the different allocation methods, a simulation model was created. The A-frame was simulated using a discrete event simulation, which meant all methods could be tested on the same data to correctly compare the performance of each allocation.

The study showed that the heuristic using association rules gave the highest number of picks for the tested period. However, it was only marginally better than the method that first removed orders that could not be picked from the A-frame and then ranked all products by their quantities sold.

The study's conclusion is that while association rule learning resulted in the highest number of picked orders, the gain of using it does not motivate its complexity. Instead a more simple approach by ranking products by their quantities sold should be used.

Acknowledgments

This master thesis was carried out at Linköping University's department of Logistics Management. The thesis concludes the author's education in Industrial Engineering and Management. The thesis was carried out at Apotea AB at their central warehouse in Morgongåva.

I, Alexander Dahl, enjoyed the problem as I could use my knowledge of computer science to solve logistical problems. I would like to thank Apotea and Maria Alriksson, my supervisor at Apotea, for giving me the opportunity to tackle this interesting problem. I would also like to thank my supervisor, Mike Malmgren, at the Logistics Management department for his support during the process. I want to thank my friends that proof read this thesis multiple times so I could (hopefully) remove all the grammar mistakes.

Contents

Abstract iii

Acknowledgments iv

Contents v

List of Figures ix

List of Tables xi

1 Introduction 1
- 1.1 Background 1
- 1.2 Aim 2
- 1.3 Research questions 2
- 1.4 Delimitations 2
- 1.5 Studied system 3

2 Problem Background 4
- 2.1 Apotea 4
- 2.2 The warehouse and material flow 4
- 2.3 A-frame 5
  - 2.3.1 The A-frame flow 5
  - 2.3.2 Replenishment 8
  - 2.3.3 The dispensing channels 8
  - 2.3.4 The conveyor belt 9
  - 2.3.5 WMS connection 10

3 Related Work 11
- 3.1 A-frame management 11
  - 3.1.1 General description and usage of an A-frame 11
  - 3.1.2 A-frame management model criteria 12
  - 3.1.3 Decision support system approach 14
  - 3.1.4 Optimization approach 14
  - 3.1.5 The multi-tier storage problem for an A-frame 15
  - 3.1.6 Order sequencing 15
- 3.2 Evaluation and simulation 15
  - 3.2.1 Discrete event simulation 16
  - 3.2.2 Conceptual model 16
  - 3.2.3 Verification and validation 17
- 3.3 Data mining 17
  - 3.3.1 Association rule learning 18
  - 3.3.2 Association rule measures of interestingness 18
  - 3.3.3 Association rule learning in warehouses 19

3.4 Safety stock . . . 20

4 Study Specification **21**
4.1 Clarification of aim . . . 21
4.1.1 Decision model . . . 21
4.1.2 Product allocation . . . 21
4.1.3 Order throughput . . . 21
4.2 Product Allocation . . . 22
4.2.1 Product allocation sub-questions . . . 22
4.2.2 Non investigated areas within product allocation . . . 22
4.3 Evaluation of an allocation . . . 23
4.4 Combination of products . . . 24
4.5 Summary of research questions . . . 25

5 Method **26**
5.1 Method overview . . . 26
5.2 Research design . . . 27
5.2.1 Case study . . . 27
5.2.2 Reliability . . . 28
5.2.3 Replicability . . . 29
5.2.4 Validity . . . 29
5.3 Pre-study phase . . . 29
5.4 Specification phase . . . 29
5.5 Modelling phase . . . 30
5.5.1 Data collection and preparation . . . 30
5.5.2 The A-frame discrete event simulation . . . 31
5.5.3 Association rules learning . . . 31
5.5.4 Allocation models . . . 32
5.6 Evaluation phase . . . 32

6 Results **33**
6.1 Data preparation . . . 33
6.1.1 Data cleaning . . . 33
6.1.2 Order consolidation . . . 33
6.1.3 Product dimensions . . . 34
6.1.4 Number of channels . . . 35
6.1.5 Normality test of product demand . . . 35
6.1.6 The maximum automation level . . . 36
6.2 The A-frame discrete event simulation . . . 36
6.2.1 Conceptual model . . . 37
6.2.2 Implementation . . . 37
6.2.3 Assumptions . . . 39
6.2.4 Limitations . . . 39
6.2.5 Verification and validation . . . 39
6.3 Association rules learning . . . 39
6.4 Allocation models . . . 42
6.4.1 Apotea's allocation . . . 42
6.4.2 Allocating by rank all orders . . . 42
6.4.3 Allocating by rank for possible orders . . . 42
6.4.4 Allocation using association rules . . . 43
6.5 Number of channels . . . 44
6.6 Allocation methods summary . . . 44

7 Discussion **50**
7.1 Results . . . 50
7.1.1 Simulation model historical data test . . . 50
7.1.2 Apotea's allocation . . . 50
7.1.3 Performance of allocation methods . . . 51
7.1.4 The maximum automation level . . . 51
7.1.5 Approximating maximum products in channel . . . 51
7.2 Method . . . 51
7.2.1 Product dimensions . . . 51
7.2.2 Hardware . . . 51
7.2.3 A-frame channel width . . . 52
7.2.4 New products . . . 52
7.2.5 Lack of prognosis data . . . 52
7.2.6 The main storage and the replenishment storage . . . 52
7.2.7 Further testing of allocation methods . . . 53
7.2.8 Selection of training and testing data set . . . 53
7.3 Research Design . . . 53
7.3.1 Reliability . . . 53
7.3.2 Replicability . . . 53
7.3.3 Validity . . . 53
7.4 The work in a wider context . . . 54

8 Conclusion **55**
8.1 Research Questions and Aim . . . 55
8.1.1 Research Question 1 . . . 55
8.1.2 Research Question 2 . . . 55
8.1.3 Research Question 3 . . . 56
8.1.4 Aim . . . 56
8.2 Recommendation to Apotea . . . 56
8.3 Future Work . . . 57
8.3.1 Product dimensions . . . 57
8.3.2 Split orders into manual and automatic . . . 57
8.3.3 Agent based simulation for the replenishment storage . . . 57

Bibliography **58**

A Technical Specifications **61**
A.1 Hardware . . . 61
A.2 Software . . . 61

B Heuristic with other minimum supports **62**

C Developed code **64**
C.1 Allocation . . . 64
C.1.1 generate_selection_from_quant_only_all_skus.py . . . 64
C.1.2 generate_selection_from_quant_only_ok_skus.py . . . 64
C.1.3 greedy_graph_search.py . . . 64
C.1.4 change_support.py . . . 64
C.1.5 draw_graph.py . . . 64
C.2 Data Analysis . . . 64
C.2.1 orders.py . . . 64
C.2.2 orders_for_date.py . . . 65
C.2.3 sku_in_channel.py . . . 65
C.2.4 article_dimensions.py . . . 65

C.2.5 find_max_automation_level.py 65
C.2.6 count_sku_support_all_skus.py 65
C.2.7 count_skus_support.py 65
C.2.8 count_sku_quantity_per_day.py 65
C.2.9 test_normality_for_skus.py 65
C.2.10 sku_in_channel.py 65
C.2.11 aframe_over_time.py 65
C.3 Machine Learning 65
C.3.1 association_rules.py 65
C.3.2 clean_rules.py 65
C.4 Simulation 66
C.4.1 aframesimulation.py 66
C.4.2 aframe_sim_by_day.py 66

List of Figures

1.1 Apotea's material flow and the studied system highlighted. 3

2.1 The material flow in the warehouse 5
2.2 The A-frame seen from the front, showing the A-shape. 6
2.3 The A-frame showing its conveyor belt, the dispensing channels and an operator stocking products in the dispensing channels. (Picture source from **Pazour2011**) . . 6
2.4 The A-frame's flow in the warehouse. Products are moved to the replenishment storage from the main storage. The A-frame's dispensing channels are filled from the replenishment storage. The WMS sends a picking order to the A-frame and an order is picked. The order is placed in a box, which can either be a small box or a big box. The box is sealed, labeled, sorted and then dispatched to the customer. . . 7
2.5 The dispensing channels eject onto the conveyor belt. The different shapes represent different products. Only one side of the dispensing channels is shown in this example. 8
2.6 The process of adding a product to a dispensing channel. In this example the channel where the product was added is too wide. The channel has to be adjusted so that the product has a tight fit. 9
2.7 The A-frame's conveyor belt. Each order will contain a number of items, shown as five blue boxes and four red boxes respectively. Each order has a picking zone, which consists of an ejecting zone where the items should be ejected to, and buffer zones where items should not land but other orders will not be affected if they do. 10

3.1 Operators refill the A-frame from the forward storage, which is in turn refilled from the main storage. 12
3.2 The state variables needed for creating a model for an automated order picking system. 13
3.3 Market baskets example. The figure has four baskets, each basket containing different items. 18

5.1 An overview of the study's method. The study had four distinct phases, Pre Study, Specification, Modeling and Evaluation. Each phase had different activities. The order in which the activities were performed is marked with arrows. 28

6.1 The many to many relationship between batch info and picking info 34
6.2 A cuboid with the faces A, B, C 34
6.3 A two dimensional projection of each face 35
6.4 Maximum automation level by date 36
6.5 A simplified conceptual model for the system 37
6.6 Flowchart for the A-frame simulation 38
6.7 Flowchart for the picking process 40

6.8 The generated graph for the association rules with different minimum supports. The nodes are blue and represent the products in the data set. The lines between the nodes are edges and represent the association rules between products. 41
6.9 Apotea's allocation 42
6.10 Allocation by rank all orders 43
6.11 Allocation by rank possible orders 43
6.12 Iterations of the heuristic 46
6.13 Allocation by greedy heuristic 47
6.14 Allocation by rank all orders with mean number of channels 47
6.15 Allocation by rank possible orders with mean number of channels 47
6.16 Allocation by heuristic with mean number of channels 48
6.17 Allocation by rank all orders with median number of channels 48
6.18 Allocation by rank possible orders with median number of channels 48
6.19 Allocation by heuristic with median number of channels 49

B.1 Allocation by heuristic minimum support = 0.0000025 62
B.2 Allocation by heuristic minimum support = 0.00002 63
B.3 Allocation by heuristic minimum support = 0.0001 63

List of Tables

6.1 Automation level by allocation and number of channels 44
6.2 Number of refills by allocation and number of channels 45

1 Introduction

In the beginning of the introduction chapter, the background of the study is presented. The background motivates the aim of the thesis, which is then introduced. To clarify the thesis' aim it is broken down into three research questions. The thesis has some delimitations which are described at the chapter's ending.

1.1 Background

Apotea AB, from now on referred to as Apotea, is an e-commerce company that specializes in pharmaceutical distribution to consumers. The Swedish pharmacy system was deregulated in 2009 (Apoteket AB 2019). Following the deregulation Apotea was founded in 2012 (Apotea AB 2019b).

Apotea has grown fast, their revenue for 2018 was roughly 2 billion SEK (Apotea AB 2019a). Because of their growth Apotea needed a bigger warehouse. In 2018 they moved into a new 38000 square meter warehouse in Morgongåva, Sweden (Apotea AB 2019b).

The warehouse's material handling has thus far been largely manual. Apotea is now looking to make investments in warehouse automation. Apotea has purchased an A-frame, an automated order picking system, as part of their automation effort. The A-frame was delivered to Apotea by the company SSI Schäfer in October of 2019. (SSI Schäfer 2019).

An A-frame is an automated dispensing machine that drops items onto a conveyor belt. The conveyor belt is divided into subsections, each representing a customer order. The control system in the A-frame is fed a customer order and the A-frame then dispenses all the order's items into one subsection on the conveyor belt. This makes the picking process in an A-frame labor free. (Bartholdi and Hackman 2019)

Apotea's A-frame has a maximum load of 1500 different products and a picking capacity of 1400 picks per hour (SSI Schäfer 2019). Apotea has roughly 18000 products and they add around 200 new products to their range of products each week. Not all of Apotea's products can be assigned to the A-frame, as they need to be of specific materials, satisfy a dimension constraint and have a specific shape. This means that a subset of possible products that can be loaded in the A-frame have to be determined. From this subset another subset has to be selected for allocation in the A-frame. Further, the A-frame can only pick complete orders, i.e. only orders where all the order's products are allocated to the A-frame. Orders that have one or more products missing from the A-frame is not processed by the A-frame, the entire order is instead picked manually.

The product allocation affects the number of orders the machine can fulfill. By increasing the number of orders that are fulfilled by the A-frame, the manual labor can be used more efficiently and give capacity for future growth. It is thus beneficial to maximize the number of orders that are fulfilled by the A-frame.

Due to the high number of possible products it is difficult for a manual planner to allocate products in the A-frame that maximizes the A-frame's order throughput. The allocation should also be frequently updated, due to changing customer demands and the ever increasing number of possible products, further increasing the difficulty of the allocation problem. (Caputo and Pelagagge 2006)

By creating a model for the product allocation decision Apotea can regularly update the allocation in the A-frame to consistently maximize the A-frame's order throughput.

1.2 Aim

To develop a decision model for product allocation in an A-frame that maximizes order throughput.

1.3 Research questions

To further clarify the desired outcome of the thesis, the aim is broken down into three research questions. They, followed by their motivation, are presented below.

1. Which criteria need to be considered for a product allocation in an A-frame?

To create an efficient model for product allocation the criteria for the product allocation need to be identified. This is done by examining and evaluating the current research on the subject.

2. How can the throughput of a given allocation be evaluated?

When creating a model it is necessary to be able to evaluate the model's effectiveness. The allocation of the generated model will be compared to the existing product allocation.

3. What combination of products result in the highest number of picked orders in the A-frame?

Apotea has a wide range of products. Customers can order any combination of items. Orders are also typically small, consisting of a few products. By finding the combination of products that have would result in the highest number of picked orders the A-frame is utilized as efficiently as possible.

1.4 Delimitations

Delimitations have been established within this thesis to create a reasonable scope for investigation. The delimitations are as follows:

- The layout of the warehouse will not be included in the model, the placement of the A-frame and its replenishment storage will be considered to be fixed.
- The changes in the manual inventory will not be considered. It is possible that some order picking cycles are affected by moving items from the manual storage to the A-frame, but that is considered out of scope for this thesis.

- The warehouse is high intensity, meaning products are generally not kept in storage more than a month. The holding cost of items in the replenishment storage and A-frame will not be considered.
- The study will not consider how a change in future demand would affect the product allocation.
- By Swedish law, purchased pharmaceutical products can not be returned. Since Apotea mostly sell pharmaceutical products they have a low rate of returns. Thus consumer returns are not considered.

1.5 Studied system

To further clarify what the thesis is studying, a simple illustration of Apotea's material flow and the studied system is seen in Figure 1.1. The study only deals with the A-frame and its direct input and output.

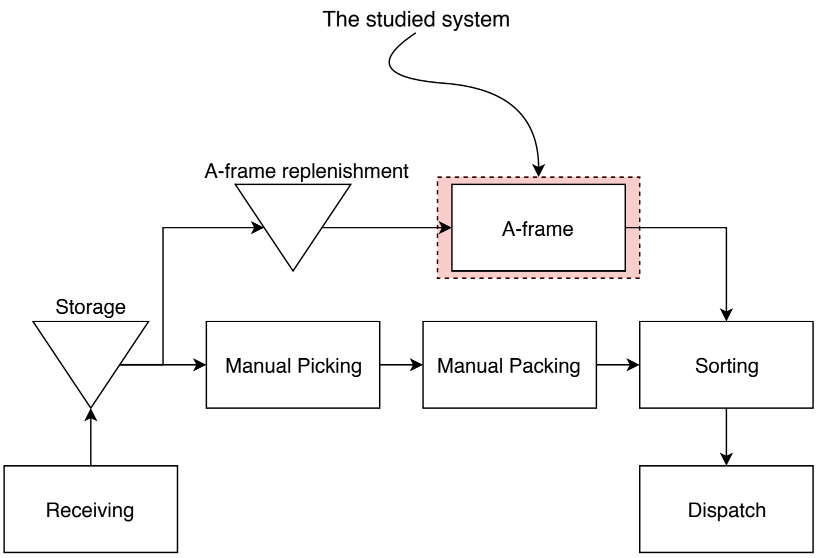


Figure 1.1: Apotea's material flow and the studied system highlighted.

2 Problem Background

In this chapter the case company, Apotea, and the material flow in the warehouse is briefly presented. Then the A-frame, its environment and its processes are described.

2.1 Apotea

Apotea AB is an e-commerce company that mainly sells pharmaceutical products to consumers. The company was founded in 2012 by Pär Svärdson, who had previously founded Adlibris AB, a swedish online book retailer. Apotea's slogan is "fast, cheap and free shipping". They sell both over the counter and prescribed pharmaceuticals. Apotea has roughly 18000 products and their product assortment grows by 200 products each week. Apotea has had a fast growth and their revenue for 2018 was 2 billion SEK. They employ around 400 people. (Apotea AB 2019a, 2019b)

2.2 The warehouse and material flow

Apotea's 38000m^2 warehouse is located in Morgongåva. The material flow in the warehouse is typical for an e-commerce company. Goods are received and put into storage. When a customer buys an order the products are picked and packed. The package is sent to a sorting station that sorts the packages depending on which forwarding agent is handling the package. The package is then dispatched.

Adding an A-frame requires changes to the typical warehouse flow. The A-frame needs a replenishment storage which is taken from the main storage. The picking and packing process is done automatically in the A-frame. Afterwards, the package will have the same process as the orders picked and packed manually. The warehouse material flow can be seen in Figure 2.1.

Apotea uses a random storage strategy. Any pallet that arrives can be placed anywhere in the warehouse where there is a free spot. When the warehouse operators store a pallet in the warehouse the pallet's placement is registered in the warehouse management system (WMS). The WMS keeps track of where all pallets are located and uses it as a basis for guiding pickers. They use a picker to stock approach for order picking, meaning the stock's placement is considered fixed and the pickers will travel to the stock's placement to pick products.

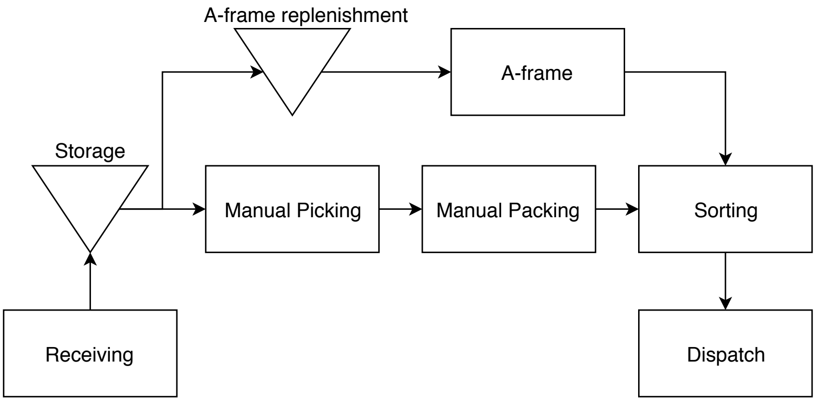


Figure 2.1: The material flow in the warehouse

2.3 A-frame

The study's aim is to develop a decision model for product allocation in the A-frame that Apotea has purchased. The A-frame and Apotea's usage of it will be described in detail. At the time of the study Apotea had not integrated the A-frame into their day to day activity completely.

2.3.1 The A-frame flow

An A-frame is an automated order picking system. In contrast to the picker to stock approach that Apotea uses in their manual warehouse, the A-frame is fully automatic, that is, the picking process is completely done by the machine. The replenishment of the machine is done by human workers.

The A-frame has dispensing channels on both sides of the machine, and a conveyor belt runs between them, making an "A"-shape, see Figures 2.2 and 2.3. These channels are refilled from the replenishment storage which consists of the products that are stocked in the A-frame. The replenishment storage is close to the A-frame so the manual operators do not need to walk far to grab items to replenish the channels. The replenishment storage is stocked from the main storage. Items for an order are dispensed onto the conveyor belt. At the end of the conveyor belt there is a double filling point, which means that items can be either dispensed into a small box or a large box. This is decided by the warehouse management system (WMS) that is connected to the A-frame. When the items have been dispensed into the box they are shaken by the machine to make sure that items are not above the box's height (called overfill), as that would block sealing of the box.

Each box leaving the machine goes to an error station by the conveyor belt. The error station checks so that no products are sticking out of the box, if any does, the box will be directed to the station to be corrected by a manual operator. The boxes are also weighed, if they are above a certain error threshold the box will be directed to the same error station and a manual operator has to check if the wrong item was dispensed into the box or if an item is missing. The operator scans the order's ID (located on the side of the box) and a software program will show the operator what should be in the order. If there is an error the order will be corrected then placed on the belt going to the the box sealing machine.

After the error station the box enters a box sealing machine that seals the box and puts an address label on it. Following the labeling, conveyor belts will bring the box to the sorting and dispatch station, which is the same as the rest of the warehouse. The entire flow can be seen in Figure 2.4.

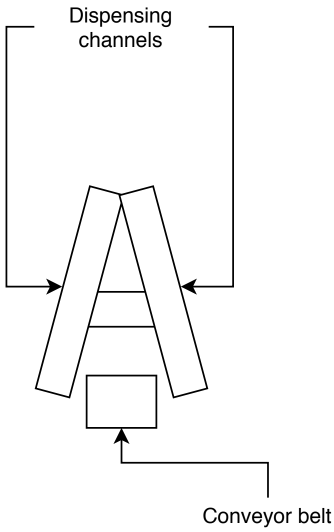


Figure 2.2: The A-frame seen from the front, showing the A-shape.

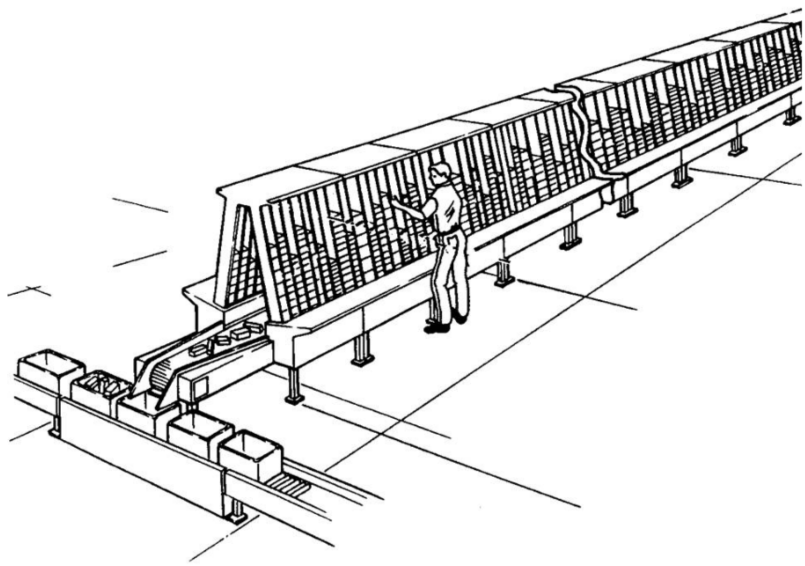


Figure 2.3: The A-frame showing its conveyor belt, the dispensing channels and an operator stocking products in the dispensing channels. (Picture source from Pazour and Meller (2011))

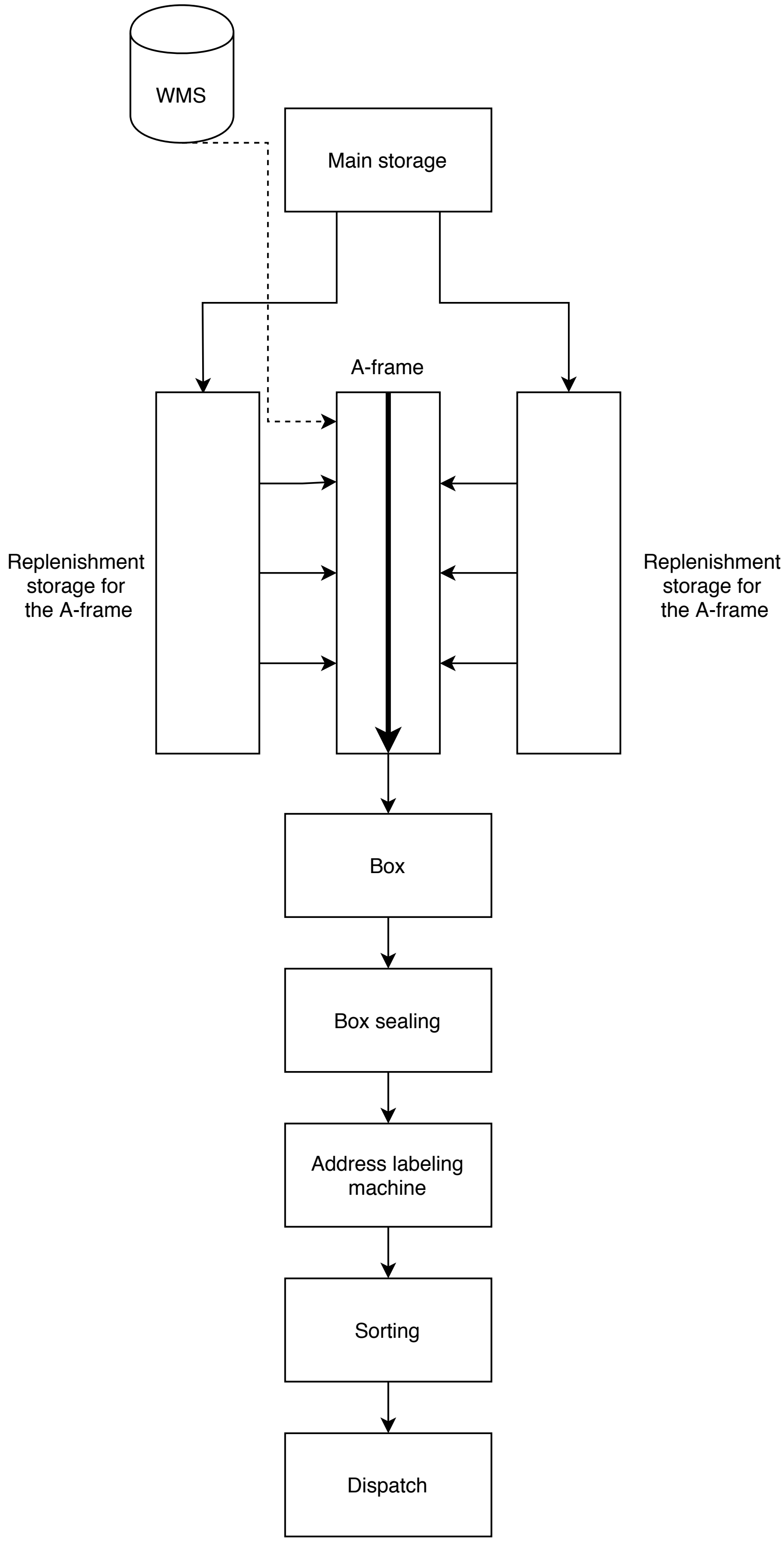


Figure 2.4: The A-frame's flow in the warehouse. Products are moved to the replenishment storage from the main storage. The A-frame's dispensing channels are filled from the replenishment storage. The WMS sends a picking order to the A-frame and an order is picked. The order is placed in a box, which can either be a small box or a big box. The box is sealed, labeled, sorted and then dispatched to the customer.

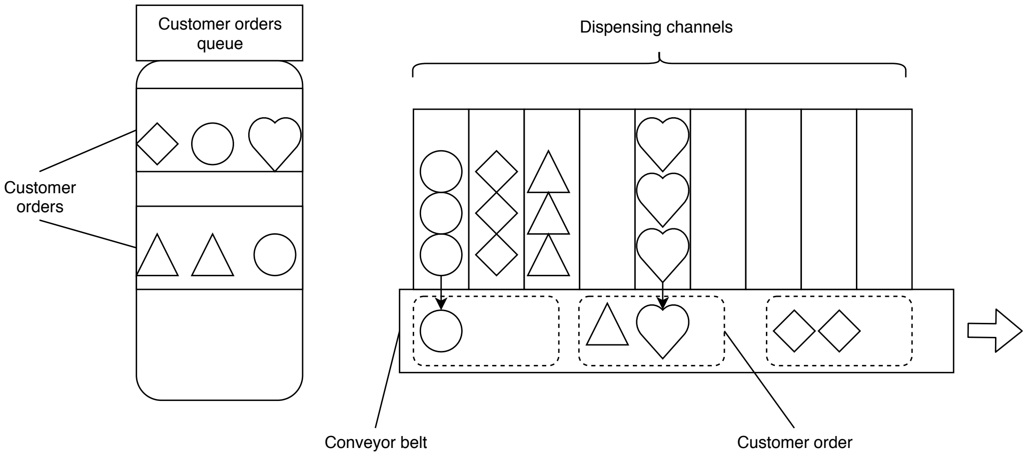


Figure 2.5: The dispensing channels eject onto the conveyor belt. The different shapes represent different products. Only one side of the dispensing channels is shown in this example.

2.3.2 Replenishment

The replenishment of the A-frame is important, if a channel stocks out, all orders that contain that specific item will be impacted. The A-frame is split up into several modules, each module having roughly 30 channels. When a channel is close to stocking out, a light on the module will turn on. This alerts a manual operator to perform a refill on the relevant channel. The operator will turn around, grab items from the replenishment and put them into the channel. Employing operators for this activity is the main operating cost of the A-frame.

2.3.3 The dispensing channels

A dispensing channel can only hold one type of product and the number of products stored in a channel depends on the dimensions of the product. A dispensing channel ejects products onto the conveyor belt when the order window passes the dispensing channel. See Figure 2.5 for an illustration. There are dispensing channels on both sides of the A-frame but only one side is shown in the figure.

The dispensing mechanic in the A-frame can not accommodate all products. What follows is a requirements list for the items that can be stocked in the A-frame:

- It has to be solid
- It has to be opaque
- It has to have dimensions within the following intervals:
  - Length: 40-220mm
  - Width: 20-120mm
  - Height: 10-100mm
  - Weight: 10-800g
- It has to be shock resistant
- It has to be stackable
- It has to be rectangular or cylindrical.

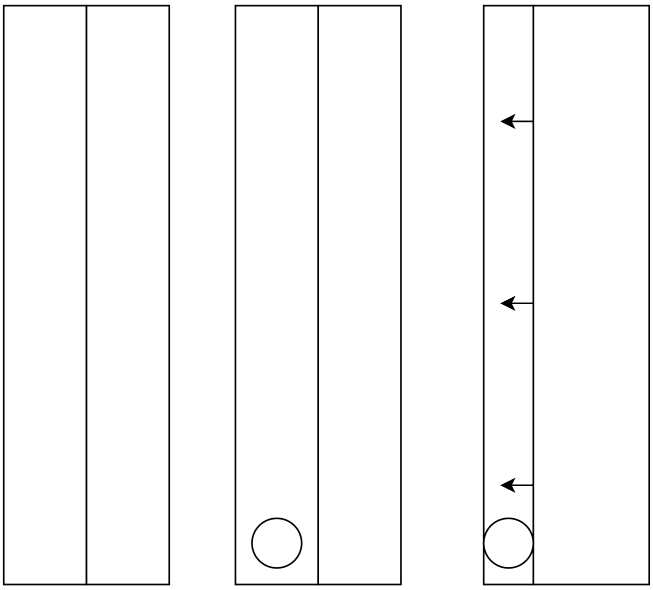


Figure 2.6: The process of adding a product to a dispensing channel. In this example the channel where the product was added is too wide. The channel has to be adjusted so that the product has a tight fit.

These requirements lower the possible number of products that can be placed in the A-frame. Apotea does not currently have data on all their products to effectively determine which products can be placed in the A-frame.

The products placed in the dispensing channels need to have a tight fit. The channels can be adjusted for this. When adjusting one channel the channels next to them are also adjusted. See figure 2.6.

Adjusting a channel is done by loosening the fastening hex key screws, pushing the channel sides so they it fits the product snugly, then tightening the screws again. This process takes a few minutes and since there are 1500 channels, adjusting all channels requires considerable effort. The manufacturer SSI Schäfer approximates that it will take two workers two weeks to completely adjust all the channels and place the allocated products in the A-frame. Because of this, products that are too small or too big can not be placed adjacent to each together, as adjusting two dispensing channels to be very close or very far way from each other disrupts the ejecting mechanism.

2.3.4 The conveyor belt

The conveyor belt runs through the A-frame. The dispensing channels eject their products onto the conveyor belt when the order window is in the correct position to do so, called the ejecting zone. In this zone all products for an order should fit. Each ejecting zone has buffer zones enclosing it. Products should not be ejected onto the buffer zone but if something is ejected onto the buffer zone the order will go to the error station and be checked by a manual operator. By having a buffer zone the other orders are not affected if a product is accidentally ejected outside the ejecting zone. The ejecting zone and the buffer zone together make the picking zone on the conveyor belt. An order that has a higher amount of products will have a bigger picking zone compared to a smaller order. The software in the A-frame calculates the picking zone length. The software is proprietary and owned by the manufacturer SSI Schäfer. An illustration of the conveyor belt's zone distribution can be seen in Figure 2.7.

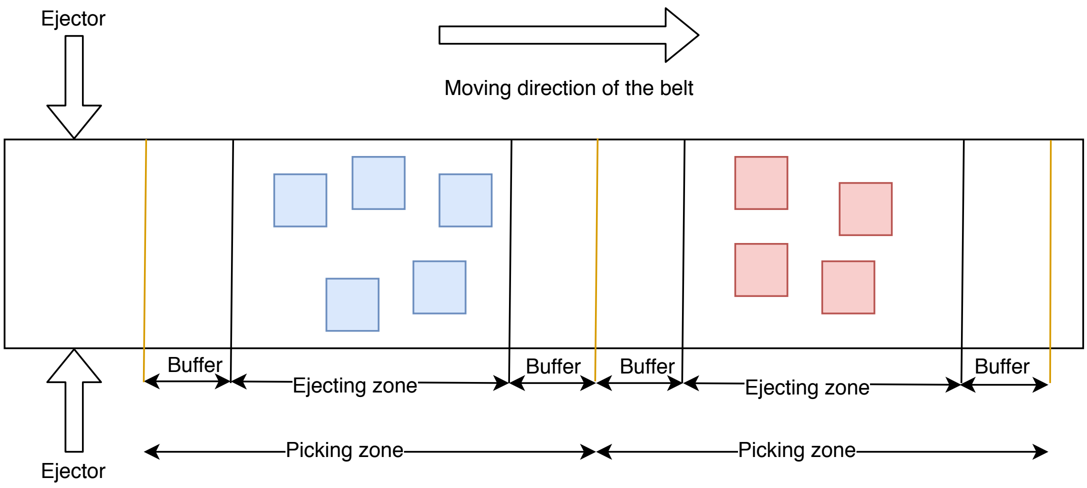


Figure 2.7: The A-frame's conveyor belt. Each order will contain a number of items, shown as five blue boxes and four red boxes respectively. Each order has a picking zone, which consists of an ejecting zone where the items should be ejected to, and buffer zones where items should not land but other orders will not be affected if they do.

2.3.5 WMS connection

The A-frame is heavily dependent on Apotea's WMS. Without a proper support from the WMS, picking orders from the A-frame would not be possible. The WMS keeps track of which channels are loaded with which products. The A-frame itself has no information of which channel contains which product. The A-frame will dispense from the channels that the WMS sends to it via a request. If a product is placed in multiple channels the WMS will distribute the picks to as many channels as possible.

The WMS decides if the order is picked and packed by the A-frame. When a customer places an order, it is sent to the WMS and the WMS will first check if the order contents are all stored in the A-frame. If they are not, the order is sent to be picked manually. If the A-frame can dispense the order contents, the dimensions of the items in the order are examined. If the order is too big for a big box then it will be picked manually. If the dimensions satisfy certain constraints then the WMS uses an algorithm that is based on the order volume and the products' dimensions to determine if they should be dispensed into a big box or a small box.

3 Related Work

This chapter presents relevant research that is related to the aim of the thesis.

The first section presents research and theories on the usage of A-frames and is related to the first research question, which criteria that need to be considered for a product allocation in an A-frame. Different approaches for A-frame management are introduced.

The second section revolves around the second research question, how to evaluate a product allocation solution. The presented research is related to the subjects of simulation and evaluation.

The third section is related to the third research question, which considers the combination of products in the A-frame. Research about association rule learning is presented. Association rule learning is a data mining method for finding interesting relations between items in data set.

3.1 A-frame management

The current research for A-frame management is presented in this section. The first research question concerns which criteria that need to be considered for a product allocation in an A-frame. To fully answer this question the identified approaches for product allocation and management of an A-frame have to be examined. In the first sub section some general information is presented that the articles have in common and then their respective approaches are introduced.

3.1.1 General description and usage of an A-frame

An A-frame is an automated order picking system that dispenses items onto a conveyor. The A-frame's control system reserves an interval of the conveyor and dispenses all the items for an order within that interval. At the end of the A-frame the conveyor deposits all items belonging to one order into a box. (Bartholdi and Hackman 2019)

Products in an A-frame must be small, be able to withstand the fall to the conveyor and they must not bounce. A-frames are generally used in cosmetic and pharmaceutical warehouses. (Bartholdi and Hackman 2019), (Pazour and Meller 2011)

According to Frazelle (2016) a warehouse typically spends 50% of its costs on picking. Since order picking with an A-frame is automated, the costs for order picking is eliminated.

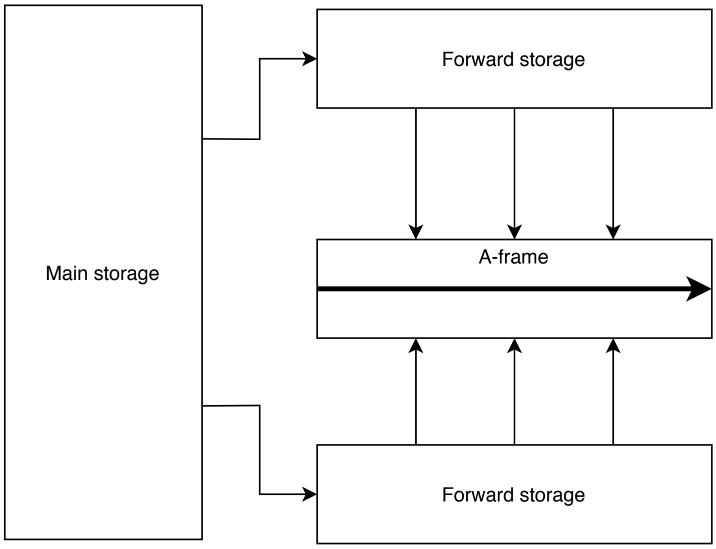


Figure 3.1: Operators refill the A-frame from the forward storage, which is in turn refilled from the main storage.

However while the order picking is labor free, operators still need to stock the A-frame's channels. Meaning the cost of operators is not completely eliminated with an automated order picking system. Further, if an order is scheduled to go through the A-frame and one item is missing then it will be forwarded to an error station where an operator need to manually pick the item and place it in the order, incurring costs. (Bartholdi and Hackman 2019), (Pazour and Meller 2011)

The dispensing channels in an A-frame can hold limited inventory, meaning they often have to be refilled. Because of the costs associated with restocking the A-frame there is normally a forward pick area close to the A-frame. The operators move product from the forward pick area and stocks them in the A-frame. The forward pick area is restocked from the main storage. An illustration can be seen in Figure 3.1. (Bartholdi and Hackman 2019), (Pazour and Meller 2011), (Caputo and Pelagagge 2006)

Automated order picking systems, such as the A-frame, are useful when the warehouse handles a high volume of small orders that need to be fulfilled within a tight deadline. (Caputo and Pelagagge 2006), (Nils Boysen 2018)

3.1.2 A-frame management model criteria

Caputo and Pelagagge (2006) performed a study on the usage of automated order picking systems at a pharmaceutical distributor in Italy. They identified a number of decisions that a manager must make in order to maximize the effectiveness and utilization of of an automated order picking system. They define all these decisions as hard and time consuming. The decisions also need to be done repeatedly and within tight deadlines.

The decisions are, according to Caputo and Pelagagge (2006):

- The selection of items to be allocated in the machine
- The storage space to be allocated to each item
- The reorder level for each item
- The maximum number of units of the same item that can be ejected for a single order

The reasons these decisions are difficult for a manual planner are, according to Caputo and Pelagagge (2006), that there is a very large amount of items involved and that customer demand is variable. They also state that the decisions heavily influence the number of orders the automated order picking system can fulfill without intervention by a manual operator. The staffing levels and operational costs are also affected.

Caputo and Pelagagge (2006) formalize their identified decisions into a set of state variables. They are presented in a list format below and in Figure 3.2.

- A = The products that will be in the A-frame.
- NC_j = The number of channels for a product j.
- LS_j = The reorder level for product j.
- Q_{maxj} = The maximum allowed quantity for a product j for one order.

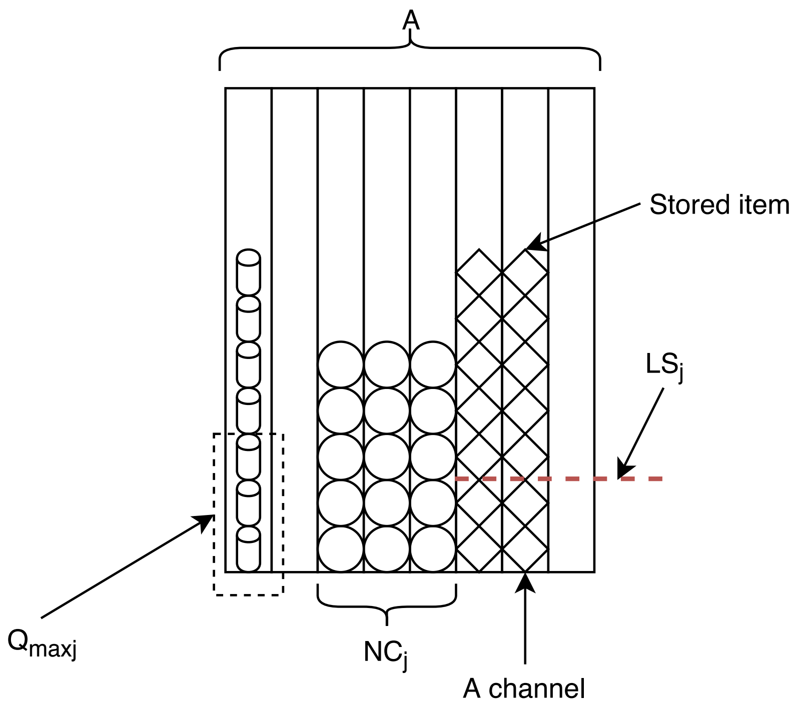


Figure 3.2: The state variables needed for creating a model for an automated order picking system.

After introducing the variables Caputo and Pelagagge (2006) explain how the variables affect the effectiveness of the machine:

The set A influences which items can be dispensed by the machine, which impacts the automation level of the system.

The number of channels NC_j allocated to each item depends on the item's turnover. A higher number of channels for an item increases the stock in the A-frame, reducing the probability of a stock-out occurring before a replenishment cycle is initiated. However, the number of channels are finite, if an item is given one more channel then another item must lose a channel, increasing that item's stock-out probability, or be removed from the machine altogether.

A lower reorder level LS_j for an item means it will be replenished less often, lowering workload for the manual operators, but it also causes a higher stock-out probability. A higher reorder level would have the opposite effect.

The variable Q_{maxj} limits the the amount of items of a product j that can be dispensed for one order. It is needed, as some large quantity orders would otherwise instantly cause a stock-out, lowering the effectiveness of the machine.

Caputo and Pelagagge (2006) continues and states that to optimize the overall machine performance the manager needs to decide the values of the above mentioned state variables. The manager needs to take into consideration the contradictory goals of maximizing the automation level, while minimizing the number of manual operators needed to restock and correct errors.

3.1.3 Decision support system approach

Caputo and Pelagagge (2006) developed a decision support system for management of an automated order picking system. They built it using Microsoft Excel and Microsoft Access. The model can simulate the effects of order picking and replenishment shifts. The simulation is based on the state variables seen in Figure 3.2.

The model developed by Caputo and Pelagagge (2006) can perform what-if analyses for system exploration by changing the state variables. When the model has been given state variables that give a proven effectiveness for the automated order picking system, the decision support system can automatically update the state variables for the next period. The main aim of the model is to set and periodically update the state variables to reduce the number of workers without increasing the stock out level. The update of the state variables is done heuristically, by modifying the variables using conditional logic and evaluating the new result.

Caputo and Pelagagge (2006) state that to get an optimal value of their defined state variables a stochastic dynamic optimization model is required. They however consider creating such a model to be unfeasible, because of the uncertainty in demand, the high number of products to be analyzed and the changing operational scenario. The authors declared this to be reason that they used a heuristic approach and not an optimization model. They deem the model to be effective despite it not being optimal, as they implemented it for an Italian pharmaceutical distributor and the model had a considerable impact on the on the distributor's cost. The model decreased the cost by 40% per picked order line compared to manual management by an experienced manager.

3.1.4 Optimization approach

The research of Pazour and Meller (2011) addresses which products should be assigned to the A-frame and in what quantity. They consider both the labor and infrastructure cost as well as the machine's order throughput. They approach the problem with mixed-integer, linear programming.

Pazour and Meller (2011) delimit their research to only consider products that would result in savings when moved from the manual storage to the A-frame. They also do not consider the products that have a very high velocity, instead proposing that these products should be placed close to the picking stations instead of in the A-frame. In their model they assume stationary demand.

The model Pazour and Meller (2011) developed includes if it is economical to invest in an A-frame and how many modules and channels the A-frame should have in their model. Their optimization model solves the allocation and assignment problem for a pharmaceutical warehouse. However, determining the throughput for a given product allocation could not be determined in a timely manner with an optimization model. Instead they developed a heuristic approach using a greedy algorithm that will find an adequate solution but not an an optimal one.

Pazour and Meller (2011) state that A-frame systems perform better when orders have a low item commonality, because the time for the A-frame to process an order depends on the number of items requested for each product. The authors also found that small order sizes in the A-frame increased both order throughput and labor savings.

3.1.5 The multi-tier storage problem for an A-frame

Jernigan (2004) studied a SKU allocation problem in a three-tier inventory system. SKU stands for stock keeping unit and is a unique identifier for a type of product in a warehouse. In a three tier inventory system the A-frame is restocked from nearby storage and that storage is in turn replenished from the bulk storage. The research is focused on minimizing the total cost of picking and the restocking storage locations in the multi-tier system. Their research is further extended to include multiple time periods meaning they also consider the cost of reallocating the system from a given allocation.

Jernigan (2004) found that by decreasing the number of SKUs in the A-frame and the number of picks the A-frame performed they could lower the total cost by 20% in a cosmetic warehouse. The warehouse they examined stored too many SKUs in the A-frame, limiting space for the most popular ones which have to be replenished often.

Liu, Zhou, Wu, and Xu (2008) also performed research on a multi-tier storage problem. In their case they had a horizontal dispenser system and an A-frame. Their research focus is to minimize restocking costs for the A-frame and the horizontal dispenser. They ignore picking costs.

Liu, Zhou, Wu, and Xu (2008) uses a heuristic algorithm for assigning SKUs to the A-frame or the horizontal dispenser. They state that the picking savings for the horizontal dispenser is greater than for the A-frame. They assign the SKUs by ranking them according to their labor efficiency.

3.1.6 Order sequencing

Boywitz, Schwerdfeger, and Boysen (2019) studied the order sequencing for an A-frame system. As the replenishment is done manually by operators, the order sequencing will affect the replenishment process. Some channels might need to be refilled at the same time with a random order sequencing, resulting in stress for the operators and possible stock outs.

Boywitz, Schwerdfeger, and Boysen (2019) developed a mixed integer optimization model for the order sequencing problem. They assume that a SKU is only assigned to one single channel. They found that by using their model their case warehouse decreased the number of unsuccessful replenishment events and required a smaller workforce of manual operators compared to a random sequence.

3.2 Evaluation and simulation

Robinson (2014) defines simulation on page 5 of their book as:

Experimentation with a simplified imitation (on a computer) of an operations system as it progresses through time, for the purpose of better understanding and/or improving the system.

Robinson (2014) states that instead of experimenting with the actual system, making a simulation and experimenting within the simulation has the following advantages:

- *Cost*, as experimenting with the real system can be costly because the day-to-day operations have to be interrupted to try the new ideas. The system would also need to be shut down while making alterations to the system.
- *Time*, as it takes a long time (weeks to months) to evaluate the experiment's performance on the real system. A simulation on a computer gives an evaluation much faster.
- *Control of the experimental conditions*, in a simulation the conditions can be fixed so the experiments can be accurately compared.

3.2.1 Discrete event simulation

According to Negahban and Smith (2014) discrete event simulation is a commonly used technique for analyzing and understanding manufacturing systems. They state that it is highly flexible and enables evaluation of different alternatives of system configurations and strategies to support decision making. In their literature study they also found that discrete event simulation is used in material handling.

Robinson (2014) define the use case for discrete event simulation as modelling queueing systems. The system is seen as a set of entities flowing from one activity to another. Each activity is separated by queues which occur when entities arrive faster than they can be processed by the next activity.

Brailsford, Churilov, and Dangerfield (2014) says that discrete event simulations consists of a number of fundamental building blocks:

- Entities, the items that flow through the system.
- Queues, where entities wait to be worked on
- Activities, where work on entities is performed
- Resources, resources have to be present to operate activities

3.2.2 Conceptual model

A conceptual model is an abstraction of a simulation model from the real world system. It is a simplified version of the real world system. Modelling the real world system in detail would take too much time and resources. But if the model is abstracted too much it would lose its value, finding a balance for the level of abstraction is important. (Robinson 2014)

Robinson (2014) says that developing a conceptual model is often seen as an art or a craft, and that there is little guidance for how to make a correct conceptual model. However, they provide a framework for developing a conceptual model.

Robinson (2014) state that the development of a conceptual model is a very iterative process. It is not possible to completely finish an activity before moving on to the next, the activities are developed in parallel. The five activities in the framework are described in the list below.

- *Understanding the problem situation:* To accurately develop a model that describes the real world, the modeller needs to have a good understanding of the problem situation. Usually the client that requests the simulation model will have an adequate understanding of the problem, but that is not always the case. The problem situation should also not be seen as static and can change as the project progresses.
- *Determining the modelling objectives:* Modelling objectives define the aim of the model. The objectives are the reference points the model will be validated against and they are a metric for the model's success. A model should be able to be used for decision-making.
- *Identifying the model outputs:* The model's output should reflect the objective of the model. The purpose of the output is to both identify if the model's objective have been achieved and, if they have not been achieved, why the objectives have not been met.
- *Identifying the model inputs:* The model inputs are key to achieving the model's objectives. The input determines the system's performance, meaning that the inputs have to be changed to find the system improvements. The model inputs are directly related to the model's objectives.

- *Determining the model content:* The model must be able to accept model input and provide the model output. The modeller must then find the interconnections between the inputs and the outputs, the model content is the layer between the inputs and the outputs. The made assumptions and simplifications about the model should be recorded and discussed.

3.2.3 Verification and validation

Stakeholders of a simulation model are concerned if a model and its result correctly reflects reality to make decisions. To find if a model is correct can be determined through verification and validation. (Sargent 2013)

For model validation there are two concepts, conceptual model validation and operational validation. Conceptual model validation validates that the theories and assumptions that builds the conceptual model are correct. Operational validation determines if the model's output has enough accuracy for the model's purpose for the domain. (Sargent 2013)

Model verification deals with the implementation and specification of the model. Implementation verification is that the simulation model has been implemented according to the specification. Specification verification verifies that the software design and the programming of the conceptual model is done in a satisfactory way. (Sargent 2013)

Sargent (2013) presents a number of ways to validate and verify a simulation model in section 4 of their article. Three of them are presented below.

- *Historical data validation*: Historical data can be collected if the simulated system exists in the real world. The data can then be split so parts of the data is used for building the model, and some data is used for testing the model to determine if the model behaves in the same way as the system.

- *Face validity*: Face validity is accomplished by asking individuals that are knowledgeable about the system if the model behaves in a reasonable way.

- *Sensitivity analysis*: Sensitivity analysis is a technique that changes the input and measures the effects of these changes. These changes should then be compared to the real system to determine if the model is the effects are reasonable.

3.3 Data mining

Today there is an huge amount of data, too much data for a human to make sense of the data without tools. Data mining is a tool for drawing valuable conclusions from data. Data mining is sometimes called knowledge discovery from data (KDD). (Han, Kamber, and Pei 2012)

Han, Kamber, and Pei (2012) define the KDD's process as the following iterative sequence:

1. Data cleaning (remove noise and bad data)

2. Data integration (combine data sets)

3. Data selection (select the relevant data)

4. Data transformation (transform data into forms that can be mined)

5. Data mining (apply methods to extract data patterns)

6. Pattern evaluation (find interesting patterns of value)

7. Knowledge presentation (visualize the data to users)

As can be seen in the sequence, data mining first occurs in step 5. The steps 1-4 are data preprocessing steps, they have to be done to prepare the data for mining. If these steps are not performed correctly, the derived conclusions can be erroneous, causing businesses to make wrong decisions. The following steps 6-7, regards the evaluation and the presentation of the found data patterns. By finding interesting patterns, meaning patterns that are valid, unique, understandable and useful, knowledge from the data can be extracted. (Han, Kamber, and Pei 2012)

3.3.1 Association rule learning

One data mining technique is association rule learning. Association rule learning identifies relationships between frequent itemsets in data. Frequent itemsets are items that frequently appear together in a dataset. By searching for recurring relationships in a data set associations, correlations and other interesting relationships in the data can be found. Businesses can use these relationships to aid them in their decision-making process. (Han, Kamber, and Pei 2012)

An application of association rule learning is market basket analysis, which was introduced by Agrawal, Imieliński, and Swami (1993). The analysis finds association rules between the different items that the customer frequently buy together, giving knowledge to business owners. Han, Kamber, and Pei (2012) gives an example, consider the customer baskets in Figure 3.3, if a customer buys milk then what is the probability that the customer also buys bread.

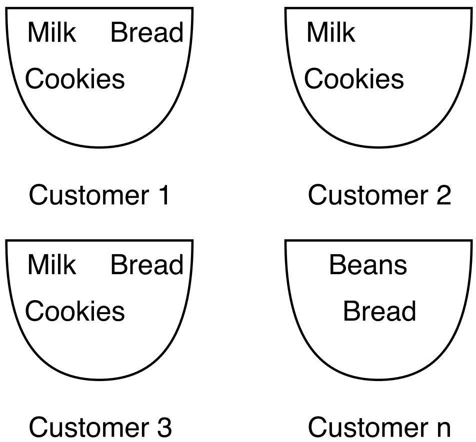


Figure 3.3: Market baskets example. The figure has four baskets, each basket containing different items.

Another example that Agrawal, Imieliński, and Swami (1993) presents is by finding the products that are frequently bought when a customers buy bagels, then the store can determine which products would be impacted if they stop selling bagels.

Two algorithms for finding association rules in a data set of transactions are the Apriori algorithm and the FP-Growth algorithm. The Apriori algorithm was introduced by Agrawal, Imieliński, and Swami (1993) and the FP-Growth algorithm was introduced by Han, Pei, and Yin (2000).

3.3.2 Association rule measures of interestingness

As stated in Section 3.3, data mining is about finding *interesting* patterns. Two measures of interestingness for association rules are *Support* and *Confidence*. They are both expressed in percentages. Han, Kamber, and Pei (2012) describe them as:

Support shows what percentage of the total item set that the association rule is true for. In the example in Figure 3.3, the rule $Milk \rightarrow Bread$ has a support of 0.5 or 50%.

Confidence is the percentage of occurrences the association rule was found to be true. In the example in Figure 3.3 the rule $Milk \rightarrow Bread$ has a confidence of 0.667 or 66.7%.

Agrawal, Imieliński, and Swami (1993) also states that support should not be confused with confidence. Confidence is the measure of a rule's strength, support is the statistical significance of a rule.

By using a minimum support and a minimum confidence thresholds, uninteresting rules can be removed. However, many of the rules generated are still not interesting to the user. Even if a rule $A \rightarrow B$ has a high enough support and confidence, there might not be any correlation between A and B. A correlation measure for rules is called *lift*. (Han, Kamber, and Pei 2012)

The lift for a rule $A \rightarrow B$ is defined as the confidence of the rule divided by the support of B or confidence$(A \rightarrow B)/$support(B). If the lift is equal to 1, then A and B are independent. If the lift is greater than 1, then A and B are positively correlated. If the lift is less than 1, then A and B are negatively correlated (Han, Kamber, and Pei 2012). In the example in Figure 3.3 the lift for the rule $Milk \rightarrow Bread$ would be $0.667/0.75 = 0.89$ meaning that the rule has a negative correlation. Continuing the example, the rule $Milk \rightarrow Cookies$ has a confidence of 100% and thus a lift of $1/0.75 = 1.33$, the rule has a positive correlation.

3.3.3 Association rule learning in warehouses

This section presents research where association rule learning has been used in warehouses. The papers are not directly related to the aim of the thesis, as they do not consider using association rule learning for allocation in an A-frame. However this shows that association rule learning can be applied to problems in warehouses.

Chen, Huang, Chen, and Wu (2005) studied the use of association rule mining for order batching in warehousing. Order batching is a process where a manual operator (picker) picks more than one order on their picking tour. They used association rule mining for discover associations between customer orders in an order database. Their approach automatically grouped orders into batches using the Apriori algorithm.

The method Chen, Huang, Chen, and Wu (2005) used for generating order batches is split into two stages. In the first stage they mine an order transaction database to find associations between customer orders. These association are expressed in support, confidence and lift. They used the following thresholds for the Apriori algorithm:

- $\theta_{support} = 0.01$
- $\theta_{confidence} = c_{avg}$ where c_{avg} is the average confidence value for the data set.
- $\theta_{lift} = 1.0$

The second stage developed by Chen, Huang, Chen, and Wu (2005) groups the orders into batches using a clustering procedure. The association rules generated in the first stage are used in the clustering. Orders with higher associations, i.e. more similar products in them, are put in the same clusters. The clustering heuristic maximizes the support value between orders for each batch. Batching orders with higher associations decreases the travel distance for order pickers. They found that their approach outperformed other heuristics on the metric of travel distance.

Li and Li (2018) extended the research of Chen, Huang, Chen, and Wu (2005) for the order batching problem. Li and Li (2018) used the FP-growth algorithm instead of the Apriori algorithm. In E-commerce customer orders are generally small batch and high frequency, leading to a database with a huge number of transactions. The implementation of the Apriori algorithm repeatedly scans the entire database, making it slow for very large data sets. They found that the FP-growth was significantly faster for the order batching problem. They tested FP-growth against the Apriori algorithm on the same data set and the FP-growth performed the same analysis in 30% of the time.

Chan, Pang, and Li (2011) studied the application of data mining on storage location in a warehouse. They used the Apriori algorithm to determine the association between products based on order history. With the associations established they assigned the products are highly correlated to be placed closely together to minimize the total travel distance for picking and put away. They found that their solution lowered the total travel distance in the warehouse compared to a random or dedicated storage approach.

Ito and Kato (2016) used the Apriori algorithm to dynamically place products in a warehouse. They used the following thresholds:

- $\theta_{support} = 0.25$
- $\theta_{confidence} = 0.4$
- $\theta_{lift} = 1.0$

They place products that correlate close to each other. They performed a simulation on their result and found that their method was better than the existing warehouse configuration, but not by a large margin. They anticipate that the effects of their method would be greater if applied to a larger warehouse.

3.4 Safety stock

According to Oskarsson, Aronsson, and Ekdahl (2013) safety stock is a method used for guarding against increases in demand. If an increase in demand is registered and the current stock can not satisfy the new demand there will be a shortage. By having safety stock this shortage can be avoided.

A method Oskarsson, Aronsson, and Ekdahl (2013) presents a formula for finding the safety stock (SS) dimensions' is based on the standard deviation of demand (σ_D), and a safety factor (k). The formula assumes that the variation in demand follows a normal distibution. For example, a safety factor of 1.28 would result in a service level of 90%. The service level is defined as the how often there will be no shortage in the stock levels.

The formula can be seen in Equation 3.1.

$$SS = k\sigma_D \tag{3.1}$$

4 Study Specification

The purpose of this chapter is to clarify and specify the study. The study's aim is:

To develop a decision model for product allocation in an A-frame that maximizes order throughput

In the introduction 1.3 three research questions were presented. This chapter will thoroughly examine these questions and explore their sub questions.

4.1 Clarification of aim

The key words in the aim are identified and explained below.

4.1.1 Decision model

A decision model is a model that is used as a tool for decision making. Apotea wanted a decision model because their business is in a dynamic environment, meaning they have to make new decisions often. A decision model can be used not only in the present situation, but it can be updated due to changing requirements by inputting new data into the model. A decision model also requires evaluation to confirm that the model is truthful to reality.

4.1.2 Product allocation

In this context allocation is defined as selecting which items to assign to the A-frame and what number of channels they should have in the A-frame. The product allocation influences the throughput of the A-frame, as which products are assigned to the A-frame will directly influence which orders can be picked from it. The number of channels a product is allocated to will influence how often there will be a stock out in the machine, impacting throughput.

4.1.3 Order throughput

Order throughput is how many orders that can be fulfilled by the A-frame. The value or the size of the orders are not considered. This effectively means that smaller orders will be prioritized for picking in the A-frame. Pazour and Meller (2011) supports this statement, as they found that picking smaller order sizes in an A-frame increased the order throughput and lowered labor costs.

4.2 Product Allocation

This section is related to the first research question:

Which criteria need to be considered for a product allocation in an A-frame?

Apotea wanted to find a solution to the product allocation problem but they had no opinion on which method should be used. That made it necessary to investigate the current state of the literature to find possible product allocation methods. The literature review also helped specify which questions have to be answered to find a product allocation that maximizes the order throughput. The literature review for the A-frame can be found in 3.1.

From the literature review a number of questions were identified that needs to be answered to find a solution for the product allocation problem. Areas that were not be examined in this study were also identified. These areas will also affect a solution to the product allocation problem, but were considered out of scope.

4.2.1 Product allocation sub-questions

The sub-questions were based on the decision criteria that Caputo and Pelagagge (2006) examined. Their research is presented in 3.1.2.

Not all products can fit in the A-frame's dispensing channels. They can not be too big, too small or have arbitrary dimensions. Before deciding which products should be in the A-frame, it must first be decided which would fit in the A-frame. This is the first sub question:

Which products can physically fit in the A-frame?

After determining which products can fit in the A-frame, the selection of products to be placed in the A-frame can be considered. To make a valuable selection the popularity of the products and what combination of products are useful have to be considered. This is the second sub question:

Which products should be in the A-frame?

It is assumed that when a product is placed in the A-frame it will completely fill the channel it is placed in. There is no gain to not fill the product channel since holding costs are not considered.

To avoid out of stocks, warehouse workers will need to be dedicated to the job of filling the dispensing channels. If an out of stock for a product occurs it will impact the order throughput.

When a product is assigned more channels it implies that there will be reduced space for the other products. When adding channels to a specific product this means that some product will not fit in the A-frame at all. This is the third sub question:

How many channels should a product be allocated to?

4.2.2 Non investigated areas within product allocation

The research surrounding A-frames state the some areas are important for the product allocation problem. Some will not be covered in this study and it is problematized here.

The research by Jernigan (2004) and Liu, Zhou, Wu, and Xu (2008) considers the multi tier storage problem, see section 3.1.5. They both focus on minimizing the total operating costs for a warehouse with an A-frame. The problem of moving products to a forward storage is important and will influence the costs for the warehouse. However this problem will not be considered in this study, as the aim of the thesis is to maximize the order throughput for the A-frame.

By not having an overarching perspective the study might locally optimize the A-frame, which does not guarantee that the solution is beneficial on a warehouse level.

The order sequencing studied by Nils Boysen (2018) will affect the replenishment process for the A-frame, see section 3.1.6. They found that by creating a model for the replenishment events instead of it being random they could lower the number of stock outs in the A-frame.

The order sequencing is not considered in this study, to test and integrate this into the decision model Apotea's IT-system for the A-frame has to be modified. It is thus considered to be out of scope for the thesis.

4.3 Evaluation of an allocation

The second research question is:

How can the throughput of a given allocation be evaluated?

In order to confirm that the decision model generated by the study is correct it was necessary to perform some evaluation on it. As described in Section 3.2, one way to to evaluate a system is through simulation. The study took this approach to evaluate the solution for the same reasons that are presented in the evaluation introduction section. The same parameters, cost, time and control of experimental conditions, are described but in the context of Apotea.

- *Cost*, experimenting directly with the A-frame is not economical. The A-frame is used in Apotea's daily operations and changing the product allocation would incur significant costs each time the allocation is changed. It is better from a cost perspective to make a simulation.
- *Time*, changing the products stored in the A-frame takes a long time so it is not feasible to experiment with the real system because of time constraints. Making a simulation gives a solution faster and it can be tweaked within minutes instead of days.
- *Control of the experimental conditions*, the demand is changing from day to day, meaning it is hard to evaluate if a solution is actually good or if the consumer demands for that day was particularly beneficial for that allocation. In a simulation the testing data can be fixed so that alternative solutions can be correctly compared.

As discussed in section 3.2.2, creating a conceptual model needs to be done before the simulation model is developed. Creating a conceptual model is not trivial, the objectives, inputs, outputs and the model content have to be defined in the conceptual model. The made simplifications and assumptions also have to be stated in the conceptual model. This leads to the first sub question which is:

How should the conceptual model for the simulation be expressed?

Discrete event simulations are used for simulating queueing systems, see section 3.2.1. The A-frame can be thought of as a simple queue system. It receives a pick order from the warehouse management system, if the machine is currently not processing an order it will pick the order. If it is not free then it will place the order in a queue until it can be picked. This is the second sub question:

How can the A-frame be simulated using discrete event simulation?

4.4 Combination of products

The third research question is:

What combination of products result in the highest number of picked orders in the A-frame?

Orders in the A-frame can only be picked when all the products for an order are stored in the A-frame. This is called a complete order. If an order is picked by the A-frame and there is one or more products missing it is called an incomplete order. Incomplete orders have to be manually handled by a worker at the error station. To maximize the throughput of the A-frame there should be as many complete orders as possible.

A naive approach would be to test all possible combinations and see which combination that gives the highest number of complete orders for a test day. However this is easily proven infeasible with combinatorics. Apotea has approximately 18000 different products and the A-frame can fit 1500 different products at maximum capacity. Assume each product is only assigned to one channel. That would mean the number of combinations to test would be:

$$\binom{18000}{1500} \approx 2.06 \times 10^{2240}$$

Which is a relatively large number.

It is thus interesting to have some other approach for testing combinations. Since Apotea has operated for several years, they have data for which products customers frequently buy together. This can be used to significantly lower the number of combinations that need to be tested.

The relationships between the different products in the A-frame is useful for finding a product allocation that maximizes throughput. If for example, consumers only buy product A when they also buy product B, both product A and B need to be stored in the A-frame. If product A is stored in the A-frame and not product B, product A will never be picked. The current research on product allocation for A-frames (section 3.1) does not consider this problem.

One way to find relationships between products in consumer purchasing patterns is through association rule learning. The theory behind association rule learning is introduced in section 3.3.1.

Association rule learning needs some parameters as discussed in section 3.3.2. These parameters have to be defined correctly, otherwise the result from running the algorithm will be too large or too small and the rules might not be interesting. If the generated rules are not interesting they can not be used for finding an allocation. The first sub question is:

What parameters for association rule learning should be used?

After generating association rules they should be used to find a solution for the product allocation problem. There will most likely be more association rules than the available dispensing channels. Some ranking of the association rules has to be used to find the best possible set of products for allocation. They can for example be ranked by descending order of support, confidence, lift or a combination of these. This leads to the second sub question:

How should the generated association rules be ranked for the product allocation problem?

4.5 Summary of research questions

This section summarizes the research questions and their sub questions presented in this chapter.

- Which criteria need to be considered for a product allocation in an A-frame?
  - Which products can physically fit in the A-frame?
  - Which products should be in the A-frame?
  - How many channels should a product be allocated to?
- How can the throughput of a given allocation be evaluated?
  - How should the conceptual model for the simulation be expressed?
  - How can the A-frame be simulated using discrete event simulation?
- What combination of products result in the highest number of picked orders in the A-frame?
  - What parameters for association rule learning should be used?
  - How should the generated association rules be ranked for the product allocation problem?

5 Method

In this chapter the study's method is described. It starts with an overview of the methods used in this thesis. The next section deals with the research design, which is the framework that the study uses. The subsequent sections deals with how the actual study was carried out and are as detailed as possible to create a high replicability.

5.1 Method overview

This study had a logistical perspective, it took place in a warehouse and the aim was to increase the order throughput for a machine in that warehouse. The goal was to make some change with the management of the machine to increase the order throughput. A logistical change management methodology by Oskarsson, Aronsson, and Ekdahl (2013) is thus considered for the study and is presented below.

Oskarsson, Aronsson, and Ekdahl (2013) divide the change management for logistical projects into seven steps:

1. Clarify the prerequisites. This is an important step that finds the aim and what the studied system is.

2. Describe and analyze the current situation. This step is a basis for the next one, without an accurate depiction of the current situation it is hard to correctly evaluate an alternative solution.

3. Alternative solutions. This step is often done in parallel with step 2.

4. Compare the current situation with the alternative solutions. In this step the pros and cons for each solution are examined.

5. Choose a solution. The decision should also be put in the context of the entire organization, for example, would the most cost effective solution fit into the organization's overarching strategy?

6. Implement the solution.

7. Evaluate the result. After implementing the solution it is important to evaluate it and compare it to the past solution. It can provide insight in how well the change process was and the quality of the analysis tools that were used.

However, the presented logistical methodology revolves around changing logistics operations and does not consider how research studies should be conducted. Kothari (2004) presents a research methodology with the following steps:

1. Define the research problem. The problem might initially be very broad and ambiguous, to conduct good research the problem and its scope need be clearly defined.

2. Review the literature. To understand the problem an extensive review of relevant literature need to be conducted.

3. Formulate hypothesis. The hypothesis helps the researcher to delimit the research.

4. Formulate research design. The research design's purpose is to build a framework for the process of collecting the evidence to find a solution to the research problem.

5. Collect data.

6. Analyse data.

7. Interpret and report.

By combining the research methodology by Kothari (2004) and the logistical methodology by Oskarsson, Aronsson, and Ekdahl (2013) the thesis gets a fundamental research basis with a logistical focus. The methods are used together to define the phases of the project, which can be seen in figure 5.1. Each phase is then described further in the later sections of the chapter.

5.2 Research design

According to Bryman and Bell (2011) and Kothari (2004) a research design provides a framework for the collection and analysis of data. In this section the research design of the study and the criteria for research evaluation, reliability, replicability and validity are presented.

5.2.1 Case study

Bryman and Bell (2011) specifies case studies to be studies that have one of the following characteristics:

- The study studies a single organization
- The study studies a single location
- The study studies a study
- The study studies a single event

The thesis studied a specific problem, product allocation in an A-frame, at a specific organization, Apotea. The study can be seen as a case study because of these attributes. Bryman and Bell (2011) calls this an intensive examination of a single case. They state that the main concern with such an examination is the generated theories from the case. If the theories can be generalized to other cases is an obstacle with case studies. They also state that case studies are well suited for hypothesis testing, where the researchers have a theory before starting the study and then test the theory with a case study.

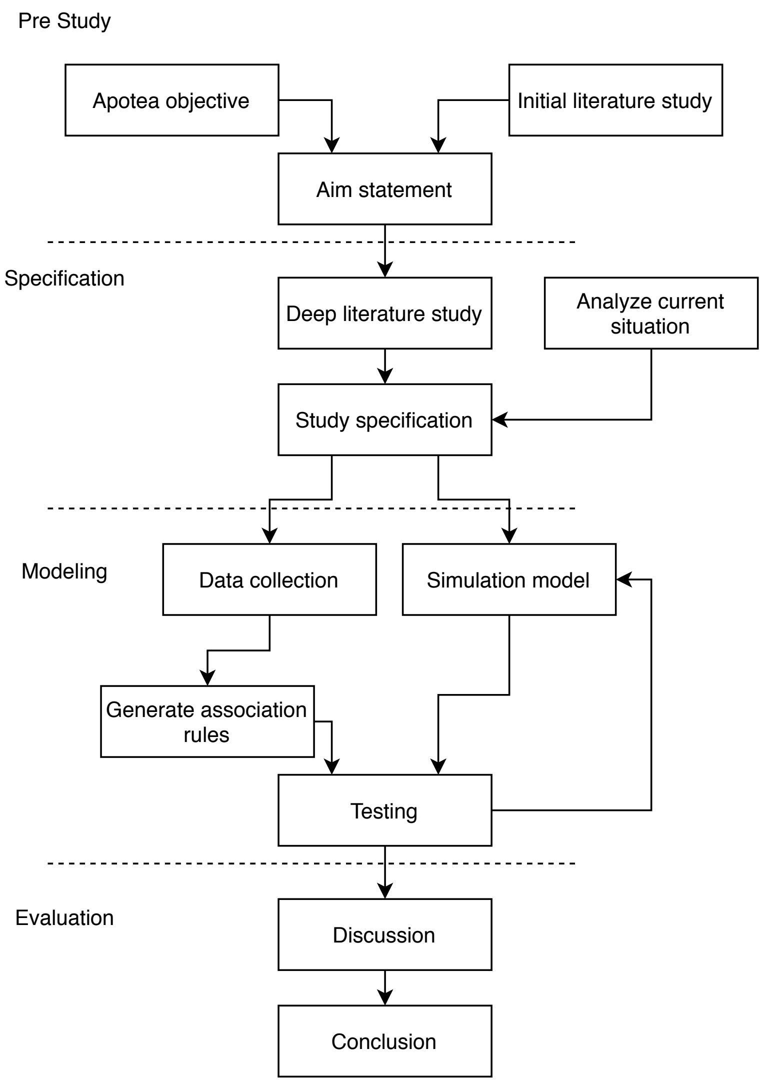


Figure 5.1: An overview of the study's method. The study had four distinct phases, Pre Study, Specification, Modeling and Evaluation. Each phase had different activities. The order in which the activities were performed is marked with arrows.

Bryman and Bell (2011) states that in case studies both quantitative and qualitative research approaches can be used. This study has an almost exclusive quantitative approach. The quantitative data is the order transactions from consumer orders at Apotea. Some qualitative data exists, and was gathered to find the aim of the study and understanding the current situation at the company.

5.2.2 Reliability

A study's reliability is how well it can be repeated and give the same result. Reliability is especially important for quantitative research. One measure of reliability is stability, which considers if the research is stable over time. The stability measure is how well the research can be tested again after some time and give the same result. (Bryman and Bell 2011)

5.2.3 Replicability

Replicability is closely associated with reliability. To be able to correctly assess the reliability of a study it has to be repeated by someone else. To achieve a high replicability the study must describe the study's procedures in great detail. (Bryman and Bell 2011)

5.2.4 Validity

Validity is a measurement of the integrity of the research, i.e. if the research measures what it was set out to measure. Validity also relates to the conclusion of the research, if the study found a relationship between two variables, can the study find a causality between them, or is there some other non identified variable influencing this. There is also the issue of external validity, which means if a study can be generalized outside of its research context. (Bryman and Bell 2011)

5.3 Pre-study phase

The study started with Apotea's problem. They had bought an A-frame to automate their warehouse. However, they were not sure how to manage it effectively. They wanted help on how they should use the A-frame in the most effective way.

This problem statement was very broad, and had to be narrowed down. This step is present in the methodologies of Oskarsson, Aronsson, and Ekdahl (2013) and Kothari (2004). An initial literature study of A-frame management and general warehouse management was conducted. This initial literature study and Apotea's problem statement made it possible to delimit the research. After delimiting the research the aim of the thesis emerged. Apotea did not prioritize cost reductions, their goal was to get as many order picks from the A-frame as possible. They also wanted some way to update the allocation in the machine as consumer demand changes. This lead to the inclusion of developing a decision model in the aim. Thus the aim of the thesis was defined as:

To develop a decision model for product allocation in an A-frame that maximizes order throughput.

5.4 Specification phase

The specification phase started with a deep literature study, which Kothari (2004) considers necessary in a research process. The search for literature was carried out by using UniSearch provided by the Linköping University library and Google Scholar. For papers that were deemed interesting their cited sources were also examined. When using Google scholar one can see which other papers have cited an article, and this was also used for finding new papers.

Articles discussing the A-frame system and its usage in warehouses was read and reflected on. From the literature study it became clear that the thesis' aim had not been covered in previous research. The examined research considers the A-frame from a cost perspective, where the goal is to minimize the total costs for the entire warehouse, whereas this study only considers the A-frame's in- and output.

Some research that examined the product allocation problem, such as Pazour and Meller (2011), assumed a stationary demand for their model. This delimits the problem and makes it easier to solve, but it was a delimitation that could not be made in this study. Apotea has a very fluctuating demand and using stationary demand would make the solution unreliable.

Because of the lack of research in the area the literature study had to branch out and study other research areas. A hypothesis was conceived by the author, that association rule learning, which is a data mining technique, could be used to help solve the product allocation problem in the A-frame. Kothari (2004) supports formulating a hypothesis to delimit the research, to further narrow down the research problem.

Another studied research area was simulation. According to Oskarsson, Aronsson, and Ekdahl (2013) different alternative solutions should be generated and then they should be compared to the current situation. For this problem, to compare solutions correctly the A-frame has to be loaded with the different allocations and then the throughput should be measured. However this is infeasible because of the time it takes to load the machine, which would both take considerable amount of time, as specified in section 2.3.3, as well as result in down time for the machine. The motivation for using simulation can also be seen in section 4.3. The study decided to use a simulation to compare the current situation with the alternative solutions.

As already mentioned, the current situation should be described and then analyzed according to Oskarsson, Aronsson, and Ekdahl (2013). This was accomplished by observing the A-frame and its environment in the Apotea warehouse. Some data from the manufacturer was also retrieved. The author also took part in a course for the warehouse employees that educated them how they should operate the machine.

The study specification could be conducted with the current situation and the literature study done. The study specification's purpose is to clarify and specify the study further. Three research questions that would help guide the research was developed. These research questions were then further broken down into sub questions to further delimit and guide the research.

5.5 Modelling phase

In this section the steps taken in the study to arrive at the result are explained. This thesis created code for the modelling phase. All the code created by the author can be viewed in an online supplement at the address: `https://github.com/aleda145/product_allocation_aframe` The data used in the study is owned by Apotea and will not be supplied.

5.5.1 Data collection and preparation

Most of the data used in the study was extracted from Apotea's IT-system by Apotea's IT-staff. The data was in the form of text files with the exception of product fit which was in a Microsoft Excel file. The batch, order and picking info was for the period 2019-08-15 to 2019-11-15. The files contained the following:

- *Product info:* Each article in their database at date 2019-11-15 and their dimensions and weight.
- *A-frame allocation:* Apotea's A-frame allocation for the date 2019-11-15.
- *Batch info:* Each batch that was picked in their warehouse and the date and time it was picked.
- *Order info:* Each order and its link to its corresponding batch.
- *Picking info:* Each pick and its information about which order it belonged to, which product and how many of that product was picked.
- *Product fit:* Apotea had tested some of their products if they could be dispensed by the A-frame or not.

Some data was acquired through physical observations of the A-frame in use. The height of the A-frame's channels was measured with a ruler. The time between output orders in the A-frame was measured with a stopwatch. The mean of the time between orders was then calculated.

With the data acquired some rudimentary data exploration was conducted. The exploration was done using *pandas* created by McKinney (2010). pandas is an open source data analysis tool for the python programming language. The exploration gave the author an understanding of the data and how the different data files fit together. In the data exploration, the data was for example plotted in different ways to give a better understanding of the data. Data errors were found in the exploration, which would need to be corrected before doing further analysis.

When data errors were found they were corrected in the data consolidation step. The different data files were consolidated into several different data files that could be used for the next steps in the modelling phase. The data was split into two different ranges. Dates from 2019-08-15 to 2019-10-31 were considered the training data set. The training data set was used when creating the models. Dates from 2019-11-01 to 2019-11-15 were considered the testing data set. The testing data set was used to see how the created model performed. The models had no data supplied to them from the testing data set.

5.5.2 The A-frame discrete event simulation

A conceptual model for the A-frame was first created and then the model was realized as a computer program. The conceptual model's objectives, outputs, inputs and model content were identified and written down. With the conceptual model created it could be transformed into a computer program.

The study used *simpy* introduced by Meurer (2017), a discrete event simulation package for the python programming language. Simpy was used in a master thesis by Holden (2017) for simulating inventory in a supply chain. This goal is similar to the objective of this thesis, motivating the choice of using the simpy package for creating a simulation of the A-frame.

The simulation program was realized from the given data. The data does not contain everything and some data was missing, meaning the simulation has some limitations and does not completely reflect reality. Some assumptions about the A-frame was also made.

5.5.3 Association rules learning

The associations rules were generated using the python package Mlxtend created by Raschka (2018). The first decision was if the Apriori algorithm or the FP-growth algorithm in the Mlxtend package should be used. Both algorithms were tested on one day of data. The FP-growth algorithm achieved exactly the same result as the Apriori algorithm but was 6325% faster. From this brief experiment the FP-growth algorithm was decided to be used for determining the association rules for the rest of the data set.

The training data set was roughly 2.5 months of order transactions. When applying the FP-growth algorithm on these transactions the author's machine's specifications (see Appendix A) proved to be insufficient. Increasing the swap space (using the computer's secondary storage as virtual memory) to 48 GB, meaning the machine had 16GB RAM and 48GB swap, was not enough for the algorithm. The algorithm self terminated when it reached peak memory usage.

This proved to be an obstacle for the study. Some other methods were tried, such as instead splitting up the data into days and performing association rules on those days. The algorithm could generate rules for each day doing this, but it there was no obvious way to combine the different days in a statistical correct way. Further, no literature was found for this approach, leading to the method being abandoned.

The chosen approach was to first remove all orders that contained products that could not fit in the A-frame. When removing those the size of the matrix that the algorithm uses to find the rules was reduced by about 50%. This still required too much memory and the algorithm could not be completed. Products that were purchased less than a total of 25 times during the period (less than 0.5 per day), were removed from the data set and the orders that they were part of. This reduced the space the matrix needed by another 50%. The algorithm could now be run without memory problems.

FP-Growth could now also run when the minimum support was set to very low without memory problems. The minimum support was set to 0.00025%. When testing higher supports this could be done by loading the rules and selecting the ones that had a support over a specified threshold. The algorithm thus did not need to be rerun. The length of rules was set to 2, meaning only direct association from one product to another were considered. Since a subset of higher order rules also consist of rules of length 2 they would be counted twice unless they were handled in a special way. Thus groups of products that associate to other groups of products were deemed not interesting.

The generated rules were loaded into a graph data structure. The package used for drawing the graph was NetworkX, developed by Hagberg, Schult, and Swart (2008). The graphs were then visually explored. Different levels of support were tested to see how the the different products related to each other.

5.5.4 Allocation models

A number of different allocation models were created. The models had different strategies for allocating the products to the A-frame. Some models did not use association rules, instead they only used the training set. These models were deliberately simple to compare their effectiveness to the more complex model that used association rules to examine if the association rules had a significant effect.

When a model was generated it was simulated in the created A-frame simulation using the picking data from the testing data set. The generated data from the simulation was then saved and plotted in a bar chart.

First the models assumed that all products should be given exactly one channel in the A-frame. Then two other models were created that assigned each product a number of channels. The first model used the mean of product demand to assign products to channels, the second model used the median to assign products to channels. The models were then simulated again but now with also the information about the number of channels.

5.6 Evaluation phase

In the modelling phase a decision model for product allocation was presented. With the model done it could now be evaluated. The evaluation phase started with identifying interesting or unexpected results. These results were then discussed and their probable causes are identified. The method and the research design of the study are also discussed. The discussion is as critical to the study as possible.

In the study's concluding phase the research questions are answered. How the thesis fulfills the aim of the thesis is also examined. A recommendation to the case company, Apotea, is presented. In this recommendation the author suggests how Apotea should use the result, what they need to be aware of and what they should study further.

6 Results

In this chapter the result of each step in the modelling phase is presented. The data used in this chapter is owned by Apotea and is considered confidential. The study will not express any real values from the data set and will intentionally be vague when describing the data.

6.1 Data preparation

This section describes how the raw data was prepared and transformed for the subsequential steps in the result.

6.1.1 Data cleaning

Apotea's batching data had some anomalies for their date handling in the data. The batching data described the date and time when a batch order was started and when it was finished. Some batch orders had date and time from the mid 20th century. When the anomaly existed in either the picking start time or the finish time, the incorrect date was assigned the value of the correct data. This assumes that an order pick is finished the same day as it was started. When both start time and finish time were incorrect the order batch was discarded and not used in the other steps.

Some orders included products that were prescribed pharmaceuticals. Because of regulations for prescribed pharmaceuticals they need to be handled in a safe environment. Only over the counter pharmaceuticals can be stored in the A-frame. All orders that contained any prescribed pharmaceuticals were removed from the order data set.

6.1.2 Order consolidation

The files containing data about batches, orders and picking had to be merged for the next steps. The order info data file contained tables that linked the batch info and the picking info, meaning their relationship was a many to many relationship. The relationship is illustrated as Figure 6.1, which shows how the many to many relationship is represented in databases. The batch info and the picking info was merged using the many to many relationship into a single table that contained the start time, box id, SKU id and quantity. This table is from now on called *order_data*.

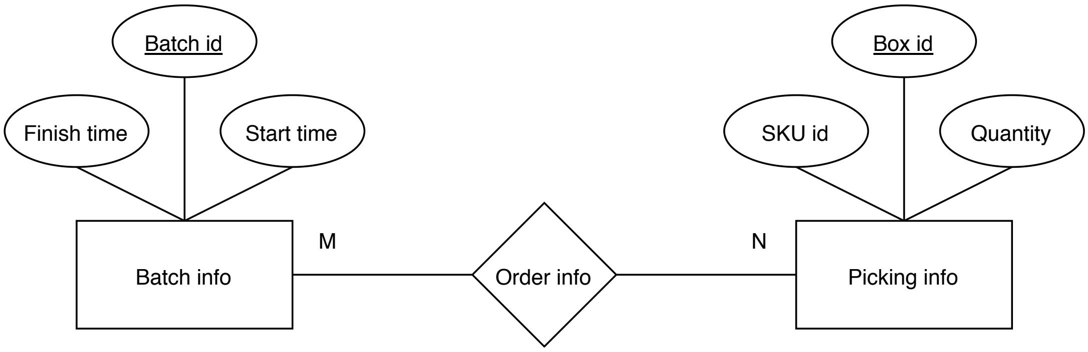


Figure 6.1: The many to many relationship between batch info and picking info

Apotea had tested some of their products to see if they could fit in the A-frame or not. If an order contained any of the items that could not physically be placed in the A-frame the entire order was removed. This operation removed roughly half of all orders. The new data was saved as a table called order *order_data_with_ok_products*.

Products that were bought less than a total of 25 times in the time period were identified. This amounted to roughly half of all of Apotea's products. The percentage of orders in the order data that contained any of these products was around 3%. These orders were removed from the *order_data_with_ok_products* and saved as a new table called *order_data_with_ok_products_and_threshold*.

For both *order_data* and *order_data_with_ok_products_and_threshold* the data was grouped by box id and the start time. The products and their quantities were aggregated into a list. They were then saved to file. They were also made into training data by removing orders that were not in the period 2019-08-15 to 2019-10-31, and saved to file.

Each day in *order_data* that was in the testing data period from 2019-11-01 to 2019-11-15 was saved to file, creating 15 files of order demand.

6.1.3 Product dimensions

In Apotea's data set for their products roughly half of their products had no dimension data. The data did not contain any of the other parameters that is stated in 2.3.3. It was not feasible to use this information to determine if a product could fit in the A-frame or not.

The number of products that could fit in a channel was calculated for the products that had valid data. The height of a dispensing channel was measured to 1700mm. A stack of products must never never exceed the height of the dispensing channel. Each product was assumed to be of cuboid shape, see Figure 6.2.

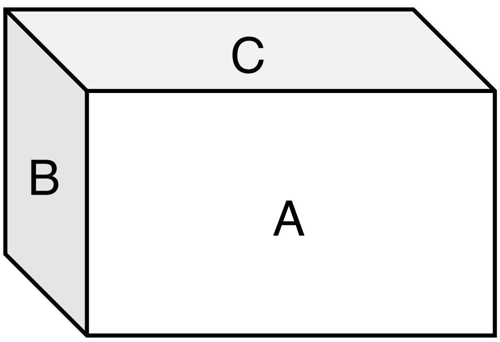


Figure 6.2: A cuboid with the faces A, B, C

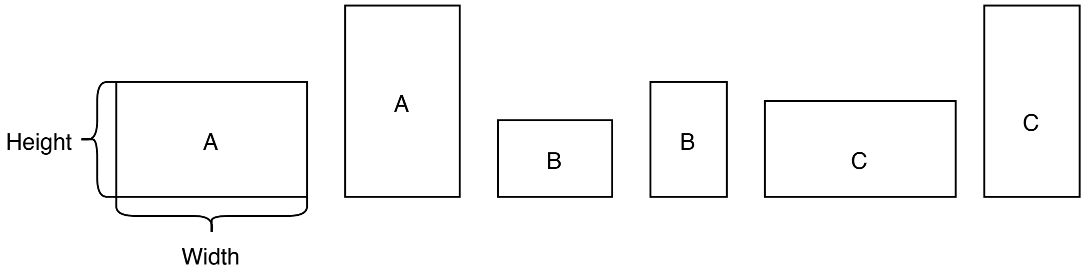


Figure 6.3: A two dimensional projection of each face

To store the maximum amount of product in a channel it should be rotated so that the face with the lowest height is placed in the channel, see Figure 6.3 for the possible rotations for a cuboid. An algorithm selected the edge with the lowest value and denoted it the height. The width was the edge with the second lowest value. The algorithm then divided the channel height with the found height for a product to find the number of times a product could be stacked in a channel. The result was rounded down to have discrete products.

Since half of the products had no dimensional data these products could not be assigned a maximum number of products for a channel. Instead the mean maximum products for all the products with data was calculated. The products that had no dimensional data was assigned the calculated mean value to allow for an approximation of the number of refills for the products.

6.1.4 Number of channels

To determine how many channels a product should be allocated to, each product's demand for the time period have to be analyzed. The number of channels a product is allocated to is the product's safety stock. It is assumed that a channel will always be completely filled with the product that is assigned to that channel, i.e. the channel will have maximum possible products in it.

A product j's safety stock (SS) will then be:

$$SS_j = n * maxstack_j$$

Where n is the number of channels and $n \in \mathbb{N}$. $maxstack_j$ is the maximum number of products that can be stacked in a channel.

To determine if a product could be assigned more than more channel the the demand for each product for each day in the training data for *order_data_with_ok_skus* was counted. If a product had a higher demand than its safety stock for one channel ($n = 1$) it was added to a list of products that had at least one stock-out. This assumes the best case scenario, that all orders in *order_data_with_ok_skus* can be picked.

Of all the products that could be placed in the A-frame, 2.9% had a higher customer demand than its maximum channel capacity at least once.

6.1.5 Normality test of product demand

As described in section 3.4 the service level safety stock method is only applicable if the demand is normally distributed. The 2.9% of products that had a higher demand than their maximum channel capacity was tested for normality. The normality test used was the normality test function from the Scipy python package (SciPy 2020). The function is based on the normality test by D'Agostino (1971).

The null hypothesis is that the demand for a product was normally distributed. The alpha value was set to 5%. The normal test generated a p value, if the p value was lower than the

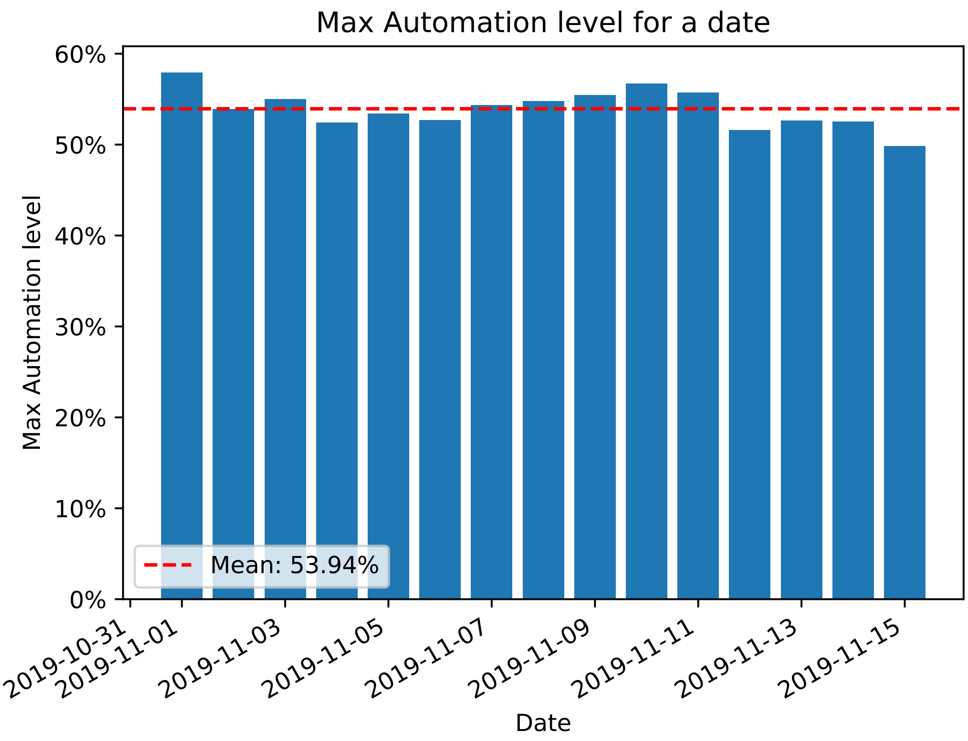


Figure 6.4: Maximum automation level by date

alpha value then the null hypothesis could be rejected. If the p value was greater than the alpha value then the null hypothesis could not be rejected.

The normality test showed that the null hypothesis could be rejected for 86% of the products and it could not be rejected for 14% of the products.

6.1.6 The maximum automation level

To have an automation level to compare to the maximum possible automation level was calculated from the *order_data*. The automation level is defined as the number of orders picked in the A-frame divided by the total number of orders.

The algorithm for calculating the maximum automation level assumes that the A-frame has an infinite capacity, i.e. all orders with products that can physically fit in the A-frame will be picked. If one or more products in an order can not fit in the A-frame then the entire order can not be picked.

The maximum automation level was calculated for each day and can be seen in Figure 6.4. The maximum automation level is an upper bound for the automation level. The level can not be higher as that would result in picking incomplete orders.

6.2 The A-frame discrete event simulation

This section describes the created discrete event simulation for the A-frame. The simulation's conceptual model will first be described, then the implementation of the simulation model is presented.

6.2.1 Conceptual model

The conceptual model was the basis for the simulation model. A simplified figure of the conceptual model that has the core attributes of the model can be seen in Figure 6.5. The conceptual consisted of the following:

- *Modelling objectives:* To test an allocation for an A-frame and measure its performance by day.
- *Model outputs:* The main output is the automation level which is defined as the number of orders that could be picked in the A-frame divided by the total number of orders for a day. The number of refills per channel and total refills are other outputs.
- *Model inputs:* The input is an allocation for an A-frame.
- *Model content:* The main model content is the orders that happened in the warehouse during a period and the process for picking these orders. The input allocation is tested for the picking process. If the products in an order can be picked with the allocation it will be picked and the number of orders that could be picked by the A-frame will be incremented by one. Which channel that picks which product and the quantity that is picked is also tracked. If an order arrives when the A-frame is busy it is placed in a queue and will wait until the A-frame is free until it can be picked.

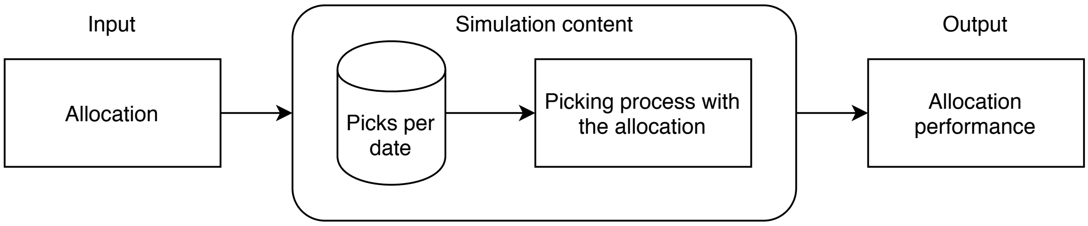


Figure 6.5: A simplified conceptual model for the system

6.2.2 Implementation

The simulation starts by loading the A-frame allocation into the simulation. It then loads all the orders for one day. It sets the day's start time as the lowest time in the order list.

For each order in the order list it evaluates if the order has all products stored in the A-frame to pick an order. If the A-frame has all the orders it will enter the simulation model, otherwise it will be logged as a manual pick.

If the A-frame is currently processing another order the current order needs to wait until the A-frame is free. The order is placed in a first in, first out queue and will wait until any order that arrived before it has been processed before it will enter the A-frame. Since the orders are all added at the same time it is necessary to check when the order is picked and the current time of the simulation. If the simulation time has not reached the time for the next order, the A-frame can not pick the order. A standby event is triggered and the A-frame will wait until the next order arrives. The A-frame simulation flow for one day of picking can be seen in 6.6.

The picking process first finds which channels the products are stored in. It will then check if there are enough stored in all the channels to pick the entire order. If there is not enough stored, it will return a failed pick and the picking process finishes. For each product in the order the channel with the most product stored is identified. Then 1 product will be picked from this channel. It checks if the channel's current capacity is lower than the specified threshold (set to 20%), if it is it will completely refill the channel. If all products have been

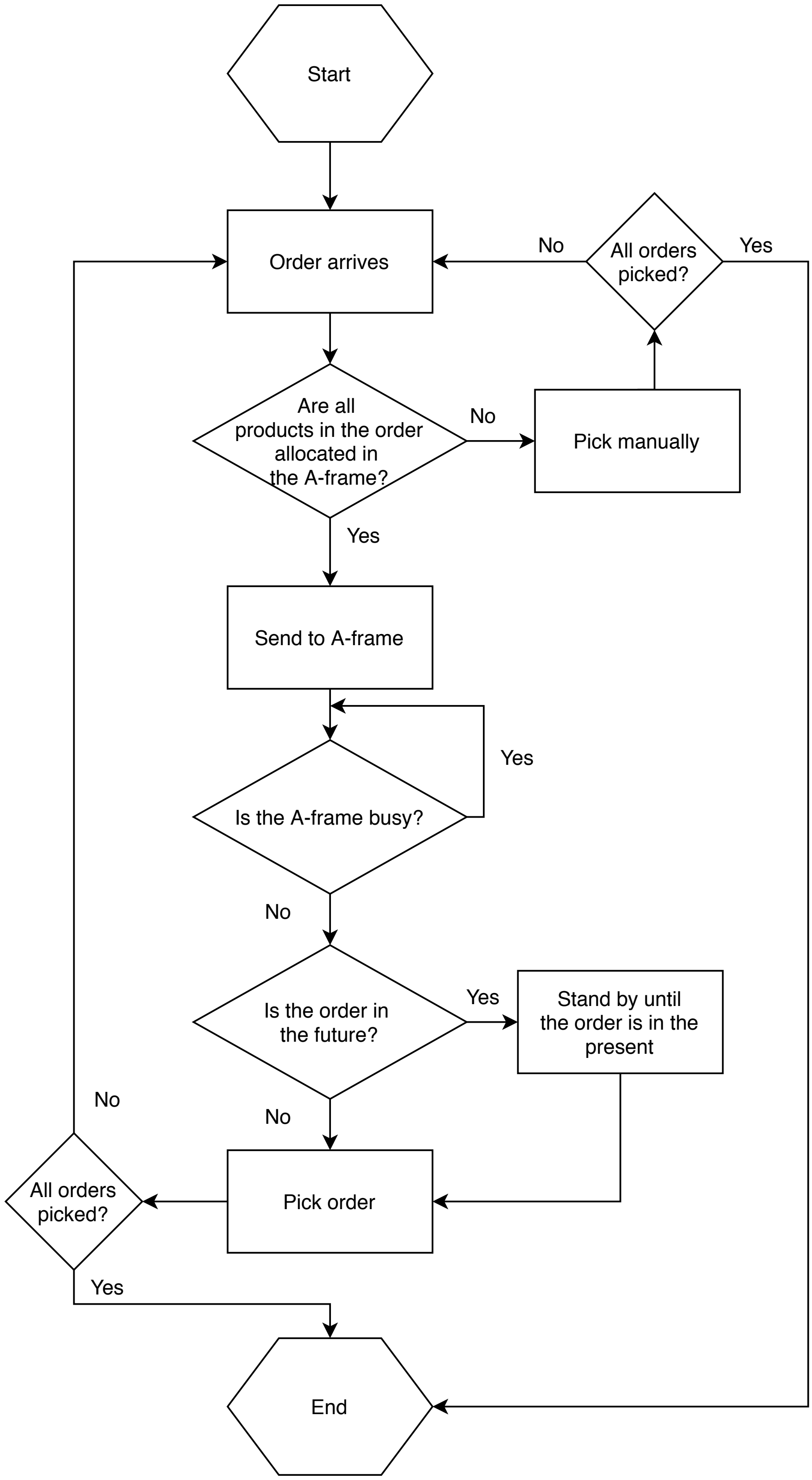


Figure 6.6: Flowchart for the A-frame simulation

picked a successful pick is recorded and the picking process finishes. A flowchart for the picking process can be seen in Figure 6.7.

6.2.3 Assumptions

There is no data of when a channel should be refilled. The simulation assumes that when a channel has less than 20% of stock in it, it will be completely refilled. The simulation assumes that no order will have its box overflow. The average time for an order to be picked by the A-frame was measured at one single date to 3 seconds. It is assumed that each order will always take 3 seconds to complete. All the channels in the A-frame are assumed to be completely filled at the start of each day.

6.2.4 Limitations

The conveyor belt is not simulated. If the conveyor belt is not running it will have an upstart time for the first order. The first order will have a time to pick that is longer than 3 seconds, and the time depends on where in the A-frame it is picked. If a product is stored at the furthest end of the A-frame it has to travel the entire length of the A-frame's conveyor belt before it is picked.

The refilling process is not simulated. In the real use case a worker will need to see if a channel is close to stocking out, then travel to the forward storage, pick up a number of products and put them in the channel. It will take some time before a channel is refilled, it is not an instantaneous event. The simulation does not keep track of the forward storage and the products there.

Jams, channel stock outs and other machine malfunctions are not simulated. These happen and workers will then correct them but the machine will have some downtime where no orders can be picked.

6.2.5 Verification and validation

As described in the section 6.2.5 finding if simulation model is correct can be done through verification and validation.

Apotea had data for which orders were picked in the A-frame for one day. A limited historical data validation was performed. This data point was tested against the simulation and the simulation could pick more orders than the historical data.

The conceptual model and the simulation was presented to management at Apotea as a face validity test. The management found the simulation reasonable.

6.3 Association rules learning

The association rules were generated after cleaning the data. The initial minimum support was set to 0.0000025. The FP-growth algorithm was used with this minimum support threshold. The rules could then be trimmed by only selecting the rules that had a support above a certain chosen minimum support value, meaning the FP-Growth algorithm did not need to be run more than once.

The initial parameters for the rules were:

- $\theta_{support} = 0.0000025$
- $\theta_{confidence} = 0$
- $\theta_{lift} = 1$

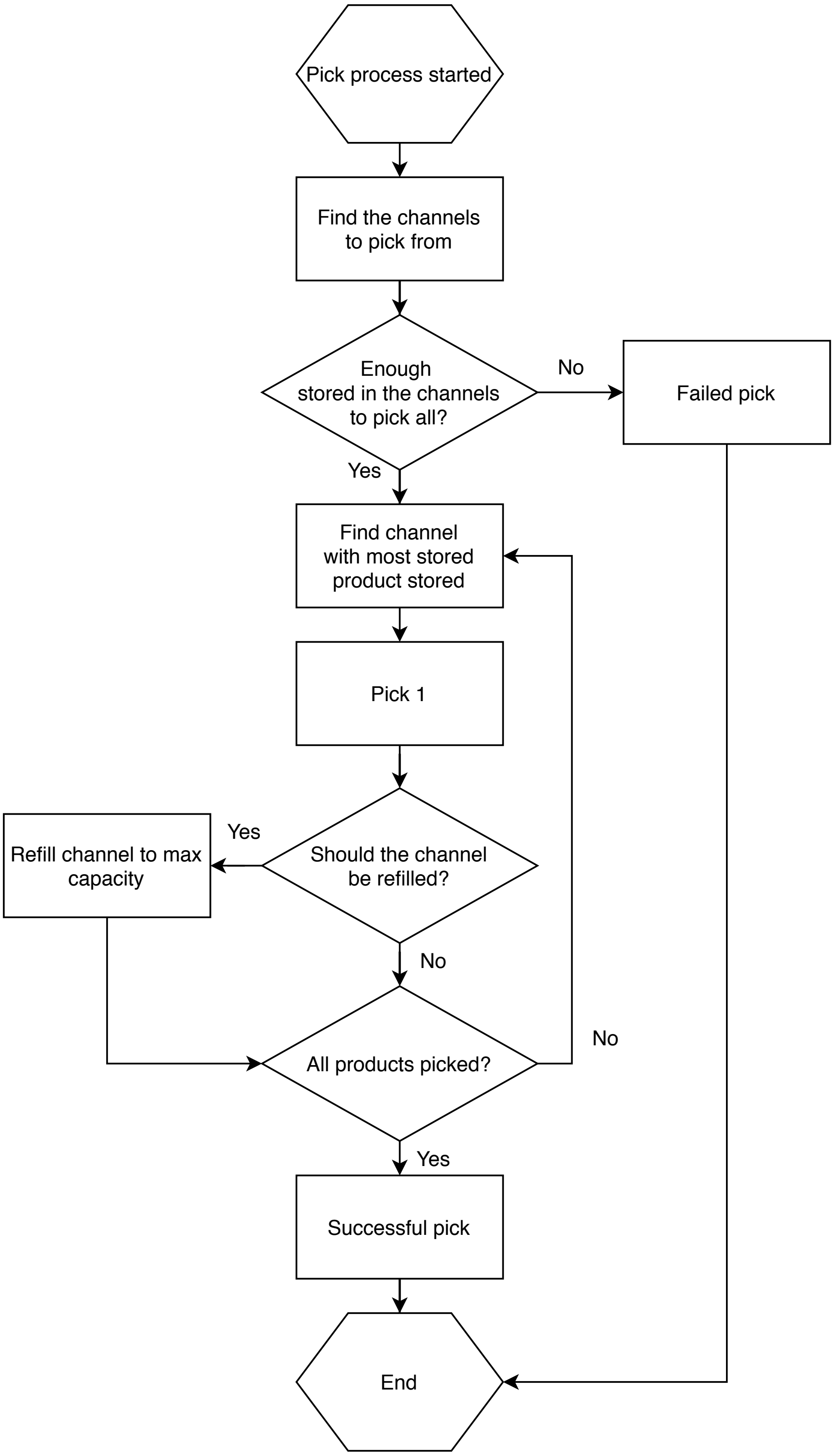
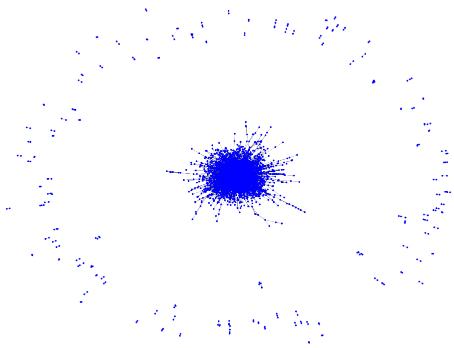
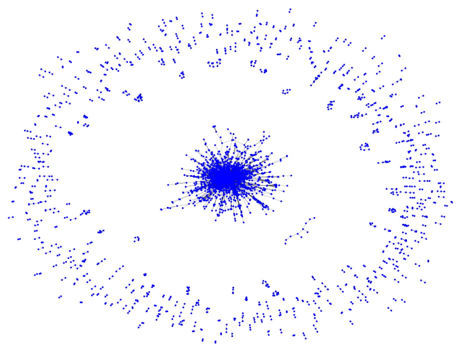


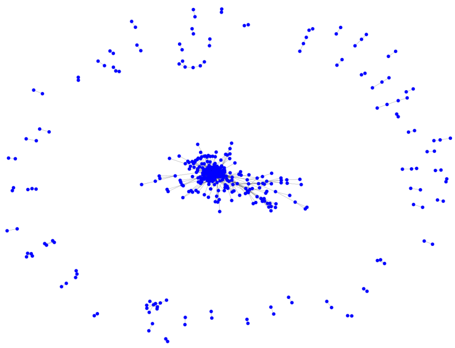
Figure 6.7: Flowchart for the picking process



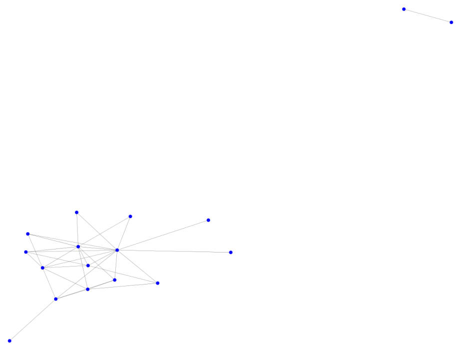
(a) $\theta_{support} = 0.0000025$



(b) $\theta_{support} = 0.00001$



(c) $\theta_{support} = 0.0001$



(d) $\theta_{support} = 0.001$

(e) $\theta_{support} = 0.01$

Figure 6.8: The generated graph for the association rules with different minimum supports. The nodes are blue and represent the products in the data set. The lines between the nodes are edges and represent the association rules between products.

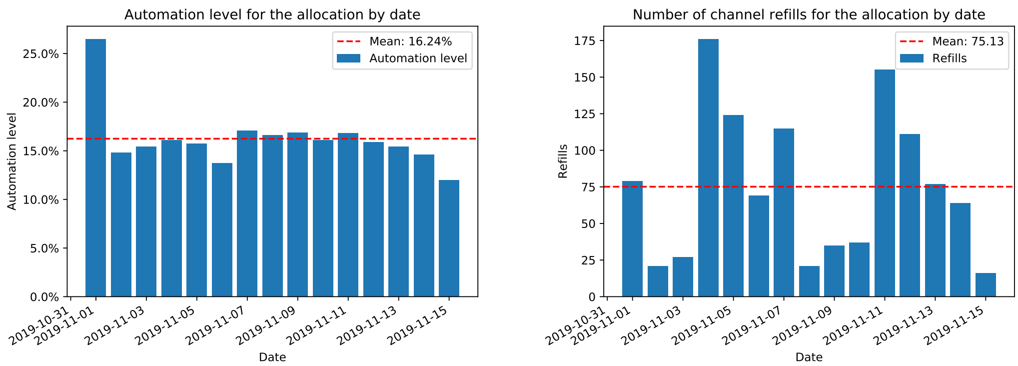
The $\theta_{support}$ was increased in steps and visualized as a graph structure. In the graph the nodes are products and the edges are assocation rules between products. The changes in the graph with the increasing minimum support can be seen in Figure 6.8. The $\theta_{confidence}$ was set to 0. The confidence metric was seen as not interesting for finding an allocation in the A-frame. With a confidence of 0 the generated rules could be treated as an undirected graph. The θ_{lift} was set to be 1. When the lift is lower than 1 the generated rule has a negative correlation, as described in 3.3.2. Only rules that have a positive correlation were considered interesting. The maximum length of a rule was set to 2. Rules of a higher length will always be a subset of a rule of length two but with a lower support if $\theta_{confidence} = 0$. Rules of a higher length were deemed not interesting.

6.4 Allocation models

This section will present the different allocation models' performance and how the allocations were generated. The training data set was incoming orders from 2019-08-15 to 2019-10-31. The testing data set was incoming orders from 2019-11-01 to 2019-11-15. The allocation's performance will be tested against the testing data set. The performance is the output from running the allocation in the simulation developed in 6.2. In this section all the generated allocations will assign one product to one channel.

6.4.1 Apotea's allocation

Apotea had created their own allocation. The allocation had not been finalized, 66% of the channels in the allocation were empty. This allocation also has some products in multiple channels. The automation level can be seen in Figure 6.9a. The total number of refills can be seen in Figure 6.9b. Apotea's method for generating their allocation was by ranking their products by number of quantities sold and using that as a guide. The product with the most quantities sold was added to the A-frame first, then the next, and so on. Products were allocated to multiple channels based on managerial experience.



(a) The automation level

(b) The number of total channel refills

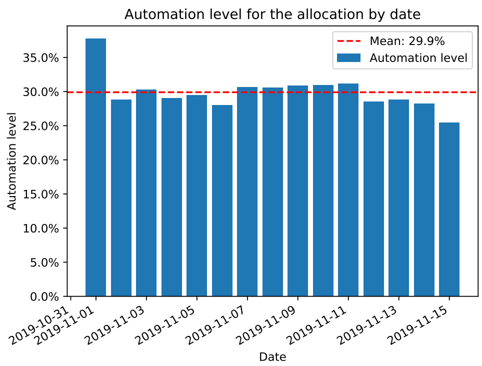
Figure 6.9: Apotea's allocation

6.4.2 Allocating by rank all orders

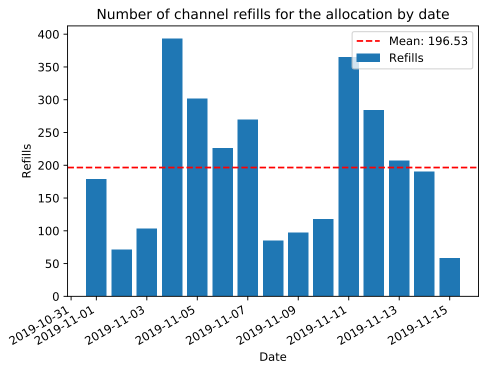
This allocation model is based on Apotea's existing method. It will count the occurrences for each product in all orders for the training data set and rank them. The product that was in the largest amount of orders will be added to the allocation, it will then be removed from the list and the next item will be added to the allocation until all the channels have been filled. If the product can not fit in the A-frame it will be skipped. This yielded the automation level seen in Figure 6.10a and the total refills seen in Figure 6.10b.

6.4.3 Allocating by rank for possible orders

This allocation model is almost identical to the one in the previous section. But instead of doing the rank on all orders only the orders that contain products that can all be placed in the A-frame are used. This yielded the automation level seen in Figure 6.11a and the total refills seen in Figure 6.11b.

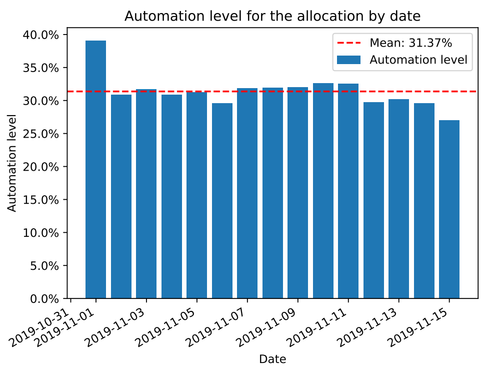


(a) The automation level

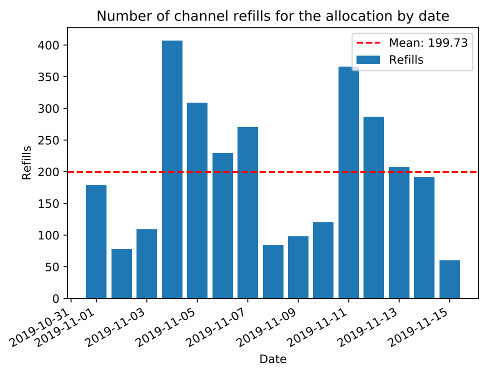


(b) The number of total channel refills

Figure 6.10: Allocation by rank all orders



(a) The automation level



(b) The number of total channel refills

Figure 6.11: Allocation by rank possible orders

6.4.4 Allocation using association rules

The association rules generated in 6.3 are placed in a graph structure. Each node is a product and each edge is an association between products, creating a graph. The support for a rule are the values placed on the edges.

By adding the support for buying a single product, i.e. how often an order only contains one single product to each node, the graph will have both nodes and edges with values. Including a node will increase the "score" for an allocation. The number of nodes are limited by the amount of channels that can be placed in the A-frame. The problem of finding an allocation is now finding a sub graph in the graph network that has the maximum score.

Finding the sub graph that maximizes the score is the following optimization problem:
x_j is 1 if a node is added to the allocation, 0 otherwise.
c_j is the single support score for a node (x_j).
$edge\_used_i$ is 1 if two nodes(x_j) with an edge between them are both active.
$edge\_score_i$ is the support between two products.

b is the maximum channel amount in the A-frame.

$$
\begin{aligned}
&\max z = \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} edge\_score_i edge\_used_i \\
&\text{subject to} \\
&\sum_{j=1}^{n} x_j \leqslant b \\
&edge\_used_i \in 0, 1 \\
&x_j \in 0, 1
\end{aligned}
\tag{6.1}
$$

Since $edge\_used_i$ depends on the values of x_j it becomes a non linear optimization problem. According to Lundgren, Rönnqvist, and Värbrand (2010) there is no general method for solving non linear optimization problems.

The problem was instead solved using a heuristic. The heuristic creates a graph network as outlined above for a given minimum support. The node with the highest single support is added to the allocation. Then the heuristic examines all nodes and adds the score for the edges that connect to the selected allocation. The node with the highest score is picked and the heuristic repeats. Since no node can be removed from the allocation the heuristic will have b number of iterations. An example of the heuristic can be seen in Figure 6.12.

The minimum support was set to 0.00001, a graph was generated and the heuristic yielded the automation level seen in Figure 6.13a and channels refills seen in Figure 6.13b. Other minimum supports were also tested. The minimum support of 0.00001 gave the highest automation level of the tested minimum supports. The other minimum supports can be in Appendix B.

6.5 Number of channels

Since the demand does not follow a normal distribution, as examined in section 6.1.5, the study can not use the safety stock service level method described in 3.4. The study tested two different ways of allocating multiple channels to products.

The first method was by taking the mean of the demand for each product, The same models as the last section are used. Each model will divide the product score with the number of channels assigned to that product. Then the models will perform the same operation as in the last section.

This yielded the automation level and number of chanel refills seen in Figures 6.14, 6.15 and 6.16.

The second method took the median of the demand for each product. The method's was approach is the same as the first method. This yielded the Figures 6.17, 6.18 and 6.19.

6.6 Allocation methods summary

A summary of the three different allocations with different channel methods is seen in the Table 6.1 and 6.2.

Allocation	1 channel	Mean channels	Median channels
Rank all orders	29.9%	18.59%	18.23%
Rank possible orders	31.37%	20.03%	19.66%
Heuristic	31.57%	20.26%	19.87%

Table 6.1: Automation level by allocation and number of channels

Allocation	1 channel	Mean channels	Median channels
Rank all orders	196.53	42.4	42.53
Rank possible orders	199.73	42.93	43.27
Heuristic	199.73	42.8	42.67

Table 6.2: Number of refills by allocation and number of channels

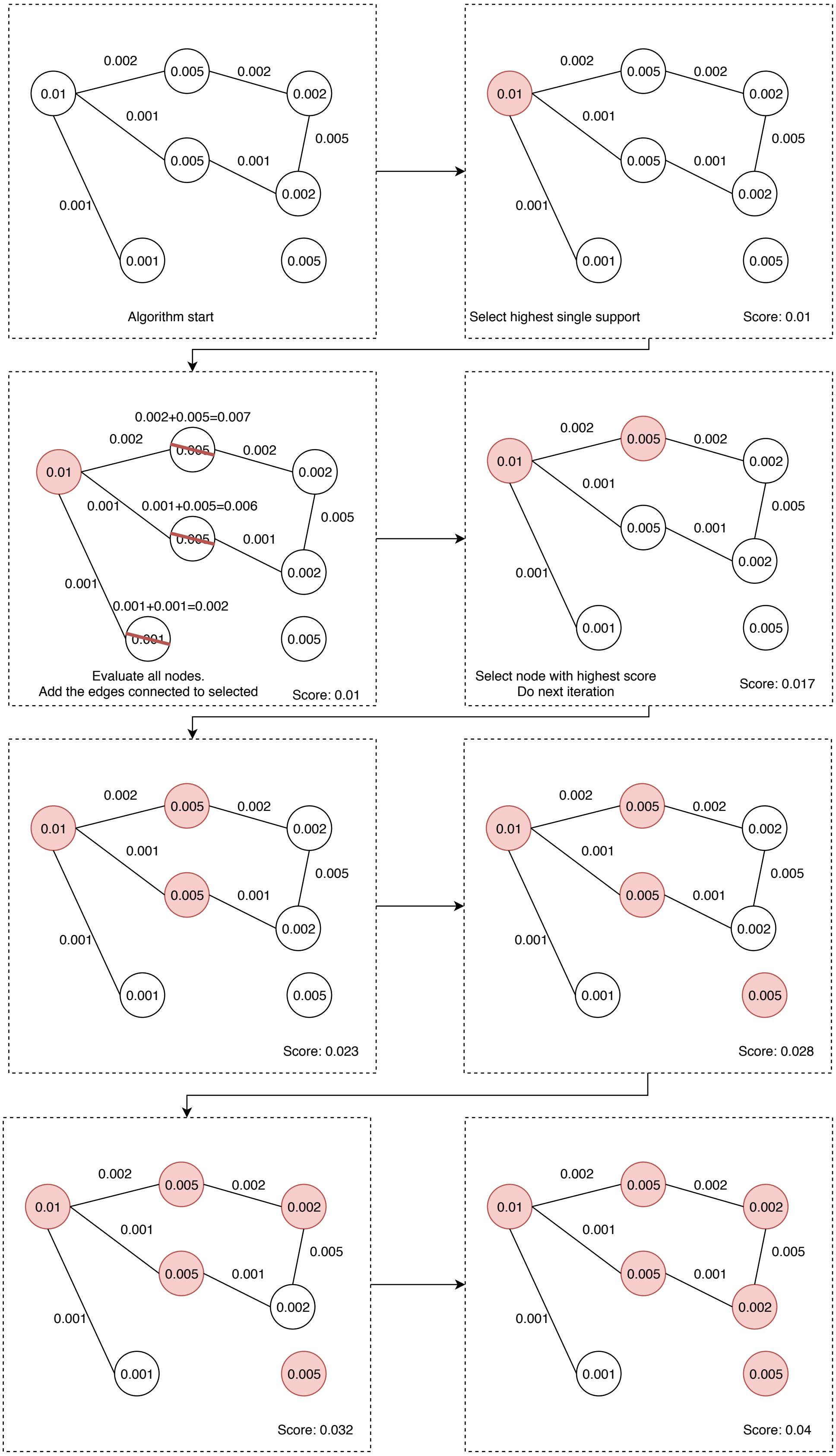
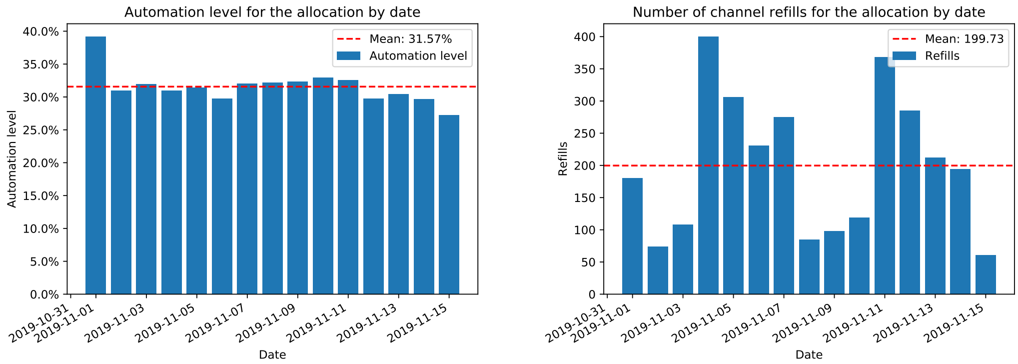


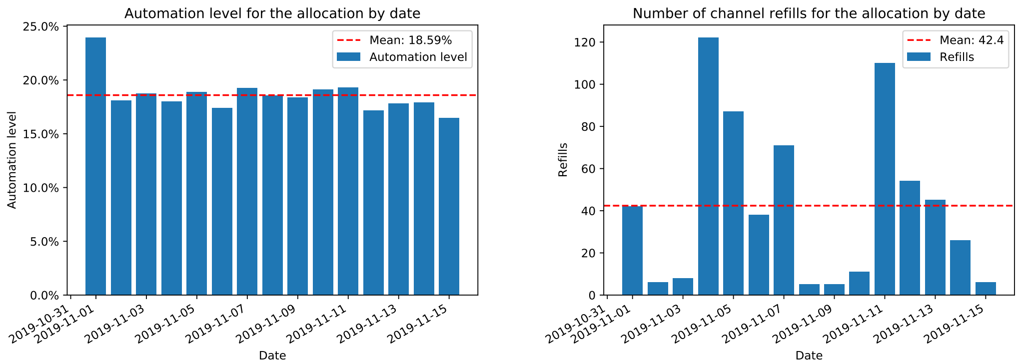
Figure 6.12: Iterations of the heuristic



(a) The automation level

(b) The number of total channel refills

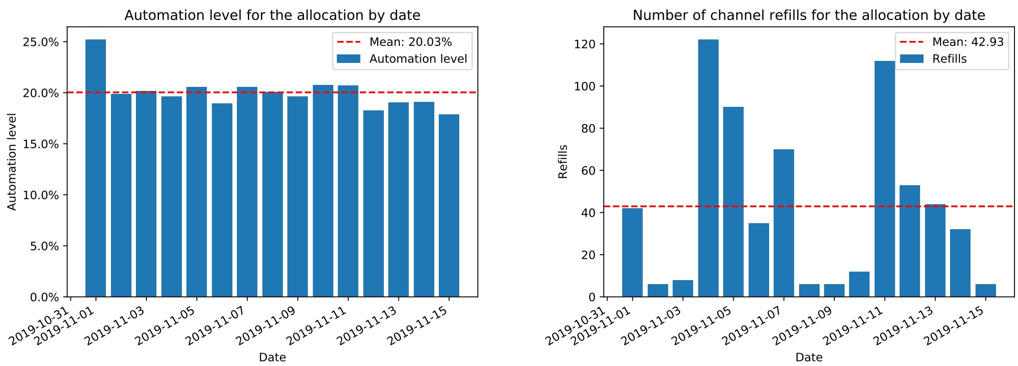
Figure 6.13: Allocation by greedy heuristic



(a) The automation level

(b) The number of total channel refills

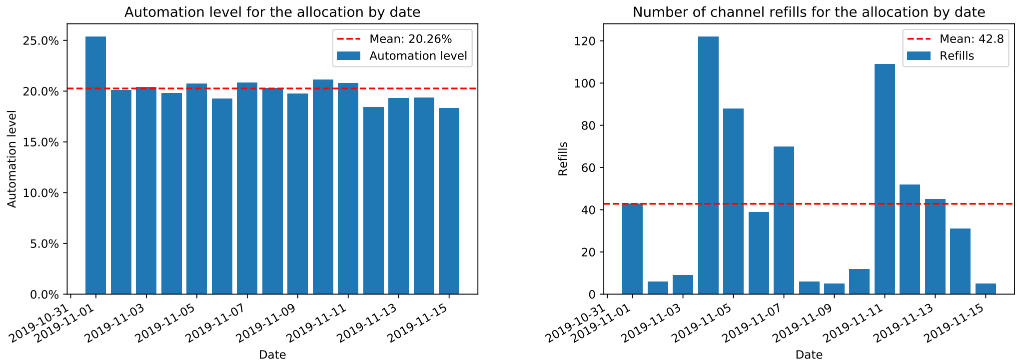
Figure 6.14: Allocation by rank all orders with mean number of channels



(a) The automation level

(b) The number of total channel refills

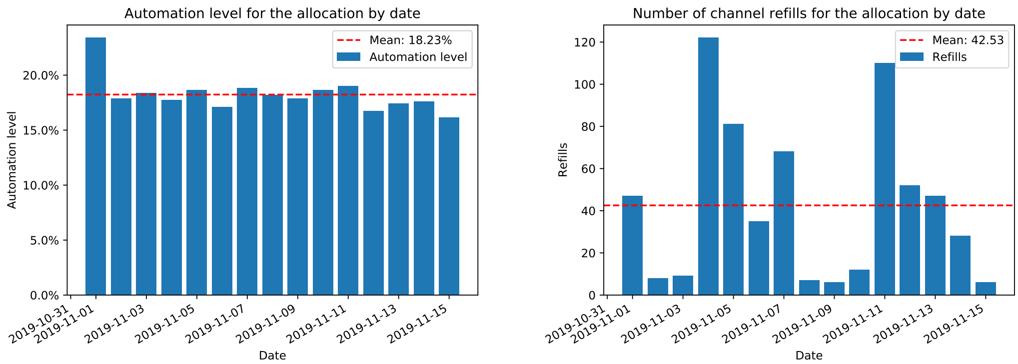
Figure 6.15: Allocation by rank possible orders with mean number of channels



(a) The automation level

(b) The number of total channel refills

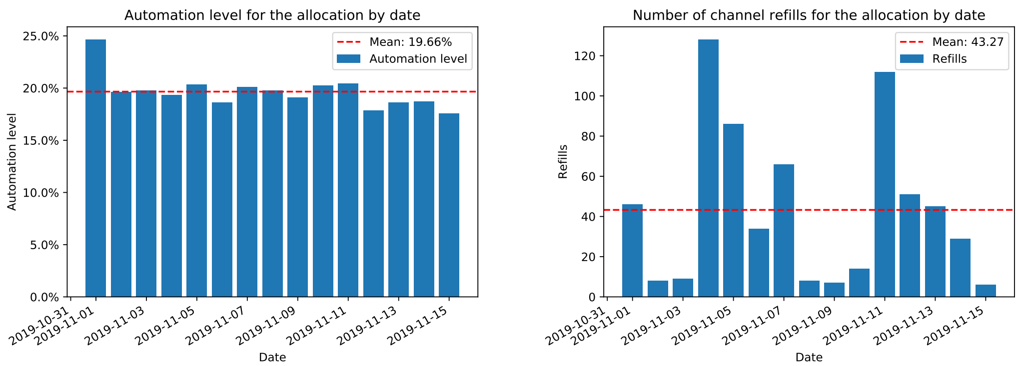
Figure 6.16: Allocation by heuristic with mean number of channels



(a) The automation level

(b) The number of total channel refills

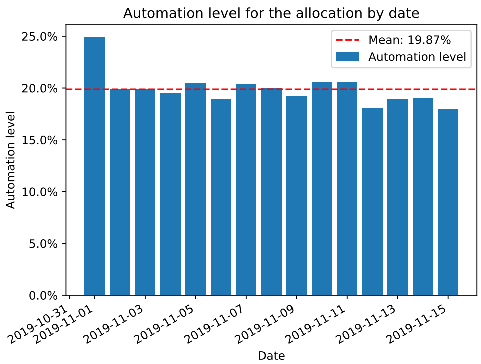
Figure 6.17: Allocation by rank all orders with median number of channels



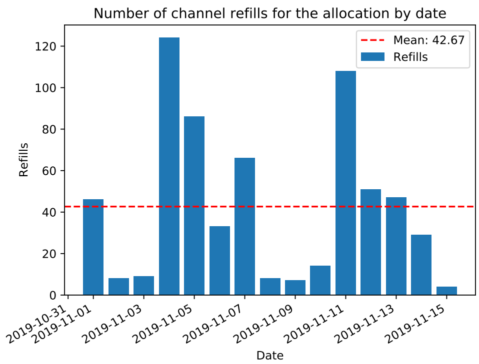
(a) The automation level

(b) The number of total channel refills

Figure 6.18: Allocation by rank possible orders with median number of channels



(a) The automation level



(b) The number of total channel refills

Figure 6.19: Allocation by heuristic with median number of channels

7 Discussion

This chapter contains a discussion about the obtained results as well as a discussion regarding the study's method and research design. The chapter ends with a discussion about the ethical and sustainable implications of the study.

7.1 Results

This section discusses the results. The areas from the result that were considered to stand out are discussed.

7.1.1 Simulation model historical data test

As described in section 6.2.5 the simulation model performed better than the actual data for the A-frame's usage. A possible reason is the assumption that the simulation model does not consider if an order would fit in the boxes or not. Apotea calculated this in their WMS before sending the order request to the A-frame as described in section 2.3.5.

The simulation model could have been verified more thoroughly by collecting more data about Apotea's current allocation and the real world picking rate for the A-frame. There was only one data point to test the simulation against the real case. The simulation model should be treated as an experimental tool, it is not necessarily a valid model. But since all allocations are simulated in the same simulation it should not affect the result of thesis.

7.1.2 Apotea's allocation

Apotea's allocation, see Figure 6.9, had both a lower automation level (16.24%) and a higher number of refills (75.13) than any of the allocation models when the allocation models accounted for the number of channels a product should be allocated to, see Tables 6.1 and 6.2. Since Apotea's allocation only utilized 33% of the channels in the A-frame it is reasonable that the automation level was lower, as fewer orders could be picked. If fewer orders could be picked then the channels would need to be refilled fewer times too, but that was not the case. Some other factor is at play than just the number of orders picked. One possible reason is that Apotea stores some bulky objects in the A-frame that have a low max channel level, meaning the channel is depleted quickly and need to be refilled often.

7.1.3 Performance of allocation methods

The heuristic with the parameter minimum support of 0.000001 had the highest automation level. However, it was not higher than the allocation by possible orders by a large margin. The gain in automation level was 0.6% when each product was assigned to one channel compared to rank possible orders. The gain was 4.9% from using rank possible orders instead of Apotea's method, rank all orders. Since the parameter minimum support has to be set for the heuristic method and there is no given method how to set this value it might lead to an allocation that performs worse than rank possible orders. The gain for the heuristic is low compared to the rank possible orders and the future effectiveness of the heuristic is uncertain. The rank possible orders is a well performing method where no parameter has to be set for using it, its only prerequisite is that the orders that can not be picked from the A-frame are removed from its ranking. The rank possible order method is a better choice for future allocations than the heuristic based on this. However, if the optimization model given in section 6.4.4 is solved then the uncertainty of guessing the minimum support would be removed.

7.1.4 The maximum automation level

The gap between the allocations' automation level, see Table 6.1 and the maximum possible automation level (53.94%), see Figure 6.4 was significant. This is expected, as with an infinite number of channels all the possible products would be put in the A-frame. However, by excluding the products that could not be placed in the A-frame the maximum automation level was only just over 50%. It is thus not reasonable for Apotea to only store some products in the A-frame and never in the manual picking storage, as roughly half of all orders would still be picked manually, and the blocking products will need products that can be placed in the A-frame to fulfill the order.

7.1.5 Approximating maximum products in channel

Half of the products in Apotea's database did not have dimensional data. The maximum amount of that could be placed in a channel for a product could not be calculated. Instead an approximation was done in 6.1.3. This approximation affects the mean refills per allocation. The study took this approach to have some value to compare to, but it is most likely wrong for most products. However, it will be equally wrong for all methods, meaning the methods can still be accurately compared. The measured performance from the methods will most likely not be accurate.

7.2 Method

This section discusses the method used. The limitations and difficulties of the study's method are identified and discussed.

7.2.1 Product dimensions

All the products were assumed to be of cuboid shape as described in section 6.1.3 since there was no data of the shape or form of any product. In the real case some products might not be able to be stacked as cuboids. Some products might also be crush sensitive, where the product that is placed at the bottom of the channel is crushed and the product ruined if the channel is filled to max capacity. The chosen method does not consider this.

7.2.2 Hardware

As described in section 5.5.3 the author's computing resources were insufficient for generating association rules using the FP-growth algorithm. Hardware could be a limiting factor if the amount of data is increased. A solution is to use some cloud provider where the amount of usable RAM is much higher.

7.2.3 A-frame channel width

In section 2.3.3 it is described how the dispensing channels need to be adjusted when a product is added to the channel. The study does not consider this. The width of the products are not considered, meaning that all the products in the generated allocation might not fit in the A-frame width-wise. The allocation is ranked from most beneficial to add to the least beneficial, if the products are added to the A-frame in that order then the automation level will not be lowered by much if the last products can't fit in the A-frame.

7.2.4 New products

If new products are introduced in November the allocation models will not consider that at all. The allocation models are based on historical data only. When introducing a new product to sell there is no data for that product to predict in how big quantities they will sell. This will affect the effectiveness of the allocations, making them perform worse. Since all the allocation models were based on historical data they will all perform equally bad for the case of introducing new products, meaning they can still be compared between each other. When introducing new products they could be predicted in some other way than through historical data. If the marketing and purchasing department has some prognosis of new products an allocation would probably perform better if that information was used in the model.

7.2.5 Lack of prognosis data

All the allocation methods only consider historical data. There was no prognosis data available for the study. This results in allocations that do not consider future demand. Some products might be seasonal, for example common cold medicine during autumn and winter, that will not be accurately allocated to the A-frame with the current methods.

7.2.6 The main storage and the replenishment storage

The study delimits how the usage of the A-frame will affect the main storage and the replenishment storage as specified in the studied system, see Figure 1.1. As the automation level was relatively high for the tested methods, see Table 6.1, this will have some effects on the other storages.

For the main storage some products will not be picked as often as before, as the A-frame will pick them instead, especially the fast moving products. The number of pickers needed will most likely decrease. Since Apotea uses a random storage strategy for their main storage the layout of the storage will not be affected. The random storage strategy in the main storage might not work well with the replenishment storage, as the replenishment storage will need to be replenished from the main storage.

The replenishment storage will be a new section of Apotea's warehouse. A major challenge is how much should of the products should be stored in the replenishment storage. If too little product is stored there the A-frame risks stocking out, if too much is stored there then the product could stock out in the main storage, affecting manual picks. This problem has been researched by Jernigan (2004).

If the study had included the main storage and the replenishment storage in the studied system, the result of the study could have been a more substantial decision model.

7.2.7 Further testing of allocation methods

This study identified three different methods for allocation products to the A-frame. These allocation methods were then each tested with three different methods for assigning the number of channels each product should have. Other methods could have been tested and the association rules could have been combined in some other way than was done in 6.4.4. If the study had tested more allocation methods it could have found a method that outperformed the developed heuristic.

7.2.8 Selection of training and testing data set

The partition of the data into a training data set and a testing data set was necessary to validate the methods. The partition itself was done arbitrarily. 2.5 months of training data and 0.5 months of testing data seemed reasonable to the author, however the partition has no scientific basis. The results could have been different if another partition of training and test data was performed.

7.3 Research Design

This section discusses the research design of the study and relates the method to the concepts of reliability, replicability and validity.

7.3.1 Reliability

This study was a case study performed at Apotea. If the study is repeated at another company using an A-frame it might yield different results. Since the allocation methods use the customer data for the allocations it depends on how customer demand varies for the other company. The heuristic might perform better if there are stronger associations between products, or it might perform worse if the associations are weaker. The number of channels the A-frame has will also vary at other A-frame implementations, this study's result might have been different if the A-frame had half the number of channels or twice as many.

If the study is repeated at Apotea, the reliability is high, the study was formed by Apotea's specifications and environment. The reliability if the study is repeated at another company is lower, however if the other company is also an online pharmacy it should yield similar results assuming that consumer patterns are similar for all pharmaceutical products.

7.3.2 Replicability

The study has specified as much as possible how the study was carried out in the method. All the code generated and used in the study is publicly available. The author's email is available in case of questions. These factors increases the replicability of the study. The data is owned by Apotea and is confidential, it can thus not be used for redoing the study using different allocation methods. This lowers the replicability of the study. Overall the study is considered to have a high replicability.

7.3.3 Validity

The study set out to create a decision model for the product allocation problem for the case company Apotea. The conclusion of the study gave a decision model that contained the data preparation steps, a discrete event simulation and three allocation models with three different methods for finding the number of channels to assign to an allocation. Apotea can extend this decision model with other parameters they consider important. From this the study is seen to have a high validity, the aim was to develop a decision model and the final result was a decision model. The external validity is lower, as the study did not consider if the decision model would be used outside of Apotea.

7.4 The work in a wider context

This sections discusses the ethical and societal aspects of the thesis.

By using an A-frame the orders are picked automatically. This means that the time it takes from a customer placing an order to it being sent from the warehouse is drastically shortened. This could change the customer's perception of how much time they are willing to wait for a package. Since the A-frame will always be quick the orders that need to be picked manually will have an increased time pressure to satisfy the customer's demands. Since the pickers in the manual storage are humans they could feel an increase in stress level if management expects them to keep up with the A-frames delivery times. Since this study's aim is to maximize the number of orders that are picked in the A-frame the study will in part be responsible for changing customers' perceptions of delivery speed, which might lead to stress workers.

With the A-frame the number of picks in the manual storage will decrease. If the number of orders decrease, less pickers are needed. If less pickers are needed the superfluous workers would be fired. This makes economical sense and that is how businesses operate, but the workers who lose their jobs will suffer.

8 Conclusion

This chapter concludes the thesis. The research questions are answered and the results relation to the aim is presented. This is followed by a recommendation to Apotea, which areas they should consider and what they should research further. The chapter ends with some areas where further research can be conducted.

8.1 Research Questions and Aim

The aim of the study was:

To develop a decision model for product allocation in an A-frame that maximizes order throughput.

Three research questions were created to guide the study into fulfilling the aim. The research questions are answered below. Then how the thesis as fulfilled the aim is presented, followed by a recommendation to Apotea.

8.1.1 Research Question 1

Which criteria need to be considered for a product allocation in an A-frame?

The main criteria for a product allocation is if a product can be placed in the A-frame or not. It proved to be unfeasible to calculate this from Apotea's data, as many parameters were missing, see Section 6.1.3. Apotea would need to gather this data for a product allocation. Another criteria is finding if a product should be placed in the A-frame or not. The study introduced three allocation methods for finding which products should be allocated to the A-frame. The number of channels each product should be assigned to was done using two methods, as presented in Section 6.5. This criteria also suffered from lack of data, as roughly half of Apotea's products had no dimensional data, meaning the maximum level of product in a channel could not be calculated.

8.1.2 Research Question 2

How can the throughput of a given allocation be evaluated?

The throughput of an allocation can be evaluated using a simulation. The A-frame was modelled as a discrete event simulation and is presented in Section 6.2. This model can be used for any generated allocation. This means different allocations can be run using the same simulation and their performance can be compared fairly.

8.1.3 Research Question 3

What combination of products result in the highest number of picked orders in the A-frame?

To find the combination of products that result in the highest number of picked orders a heuristic based on association rules was developed and is described in Section 6.4.4. This heuristic performed better than using a more simple ranking method but not by large margin. An optimization model was also introduced that if solved would result in a greater or equal number of picked orders.

8.1.4 Aim

With the three research questions answered the aim of the thesis can be presented. The decision model is a combination of the simulation and the allocation methods. Apotea can use any of the given allocation models or create their own models and use the simulation model to evaluate the result of an allocation. Apotea can use the simulation to tweak an allocation model until it performs the way that they want. Apotea's management will know which number of refills will work best for their workforce and their routines.

8.2 Recommendation to Apotea

A problem with the study was that data regarding product shapes and product dimensions was missing. If Apotea could gather this data from its suppliers or collect the data from measurements the decision model will give a better estimate of the real world. If a product can fit in the A-frame or not is also important data points and these could be measured for all products that the chosen allocation model allocates to the A-frame. Then Apotea does not need to test its entire product selection in the A-frame.

The allocation model rank possible orders is recommended to be used for the allocation problem. The developed heuristic performed marginally better but its uncertainty makes it a worse choice. Apotea should also evaluate the created simulation model with more historical data and confirm if it is effective. The algorithm that Apotea uses if a product should be picked in the A-frame should also be added to the simulation model.

Apotea should also consider the multi-tier storage problem presented by Jernigan (2004) for the A-frame. This thesis focused only on throughput of the A-frame and did not concern the problem of moving products to the replenishment storage. If the allocation model where each product is assigned to one channel then roughly a third of all Apotea's orders will go through the A-frame. This would put considerable strain on the internal logistics for the warehouse. Apotea needs to evaluate on a warehouse level which automation level is feasible for the A-frame.

The warehouse layout and Apotea's storage strategy should also be evaluated. By using the A-frame there will be flow of goods from the main storage to the replenishment storage. If Apotea continues using a randomized storage products that have a high demand in the A-frame might be placed far away from the replenishment storage. Since some operators will now need to move products from the main storage to the replenishment storage it is feasible to store the fast moving products in the main storage physically closer to the A-frame to minimize the time operators need to walk to grab products.

8.3 Future Work

This sections describe future research areas that have been identified during the study.

8.3.1 Product dimensions

Some products might be harder to refill because of their shape and size. It is interesting to rank how hard the products are to refill in the dispensing channels. If an operator is tasked with refilling only hard to refill products then they might be operating at a slower pace than the other operators. By taking this into account for the allocation methods could give a recommendation that is more aligned with reality.

8.3.2 Split orders into manual and automatic

Throughout this study the focus has been on complete orders. By removing complete orders as a constraint the customer orders could be split into automatic boxes and manual boxes. The automatic box would only contain items that can be picked in the A-frame, the manual box would only contain items that could be picked manually. The boxes would be sent separately. This would change the flow in the warehouse. The number of boxes sent would increase from the same amount of orders. The cost for forwarder agents would also increase. It is bad for the environment, as sending more boxes would require more resources. But if this means that Apotea can utilize their A-frame more effectively it might be profitable for Apotea to consider splitting the orders into manual and automatic boxes. This is a major change and should be thoroughly researched to find the total cost of the chang ein order handling. It could be another master thesis project at Apotea.

8.3.3 Agent based simulation for the replenishment storage

This study has not considered the replenishment storage and the operators that will move products from the replenishment storage to the A-frame. The number of operators needed will be decided by Apotea from managerial experience. By creating an agent based simulation of the replenishment of the A-frame, where each operator is an agent, the number of workers needed for a given allocation can be predicted. In an agent based simulation each agent will have a set of instructions and they will follow the instructions as the environment they are in changes. This can be used to test different number of operators, where they should be placed and which products should be placed next to each other in the A-frame to help Apotea decide how many workers to allocate to the A-frame. This could also be a master thesis project at Apotea.

Bibliography

Agrawal, Rakesh, Tomasz Imieliński, and Arun Swami. 1993. "Mining association rules between sets of items in large databases." In *Acm sigmod record,* 22:207–216. 2. ACM.

Apotea AB. 2019a. "Årsredovisning för Apotea AB" [in Swedish]. June.

———. 2019b. "Om Apotea" [in Swedish]. https://www.apotea.se/om-apotea.

Apoteket AB. 2019. "Monopolet avskaffas" [in Swedish]. https://www.apoteket.se/om-apoteket/apotekets-historia/ursprunget/omregleringen/.

Bartholdi, John J., and Steven T. Hackman. 2019. *Warehouse & distribution science.* https://www.warehouse-science.com/book/index.html.

Boywitz, David, Stefan Schwerdfeger, and Nils Boysen. 2019. "Sequencing of picking orders to facilitate the replenishment of A-Frame systems." *IISE Transactions* 51, no. 4 (January): 368–381. doi:10.1080/24725854.2018.1513672.

Brailsford, Sally, Leonid Churilov, and Brian Dangerfield, eds. 2014. *Discrete-Event Simulation and System Dynamics for Management Decision Making.* John Wiley & Sons Ltd, May. doi:10.1002/9781118762745.

Bryman, Alan, and Emma Bell. 2011. *Business Research Mehtods.* Oxford University Press.

Caputo, Antonio C., and Pacifico M. Pelagagge. 2006. "Management criteria of automated order picking systems in high-rotation high-volume distribution centers." *Industrial Management & Data Systems.*

Chan, Hau Ling, Anthony Pang, and Ka Wing Karen Li. 2011. "Association Rule Based Approach for Improving Operation Efficiency in a Randomized Warehouse."

Chen, Mu-Chen, Cheng-Lung Huang, Kai-Ying Chen, and Hsiao-Pin Wu. 2005. "Aggregation of orders in distribution centers using data mining." *Expert Systems with Applications* 28, no. 3 (April): 453–460. doi:10.1016/j.eswa.2004.12.006.

D'Agostino, Ralph B. 1971. "An Omnibus Test of Normality for Moderate and Large Size Samples." *Biometrika.*

Frazelle, Edward H. 2016. *World-Class Warehousing and Material Handling.* McGraw-Hill Education.

Gallagher, Sean. 2019. "Researchers find bug in Python script may have affected hundreds of studies." October. `https://arstechnica.com/information-technology/2019/10/chemists-discover-cross-platform-python-scripts-not-so-cross-platform`.

Hagberg, Aric A., Daniel A. Schult, and Pieter J. Swart. 2008. "Exploring Network Structure, Dynamics, and Function using NetworkX." In *Proceedings of the 7th Python in Science Conference,* edited by Gaël Varoquaux, Travis Vaught, and Jarrod Millman, 11–15. Pasadena, CA USA.

Han, Jiawei, Micheline Kamber, and Jian Pei. 2012. *Data Mining Concepts and Techniques.* Morgan Kaufmann Publishers.

Han, Jiawei, Jian Pei, and Yiwen Yin. 2000. "Mining Frequent Patterns Without Candidate Generation." *SIGMOD Rec.* (New York, NY, USA) 29, no. 2 (May): 1–12. ISSN: 0163-5808. doi:`10.1145/335191.335372`. `http://doi.acm.org/10.1145/335191.335372`.

Holden, Lauren. 2017. "Inventory Optimization Using a SimPy Simulation Model." Master's thesis, East Tennessee State University.

Ito, Yusuke, and Shohei Kato. 2016. "An Apriori-Based Approach to Product Placement in Order Picking." In *2016 IEEE International Conference on Agents (ICA).* IEEE, September. doi:`10.1109/ica.2016.039`.

Jernigan, Stephanie A. 2004. "Multi-Tier Inventory Systems With Space Constraints." PhD diss., Georgia Institute of Technology.

Kothari, C. R. 2004. *Research Methodology.* New Age International.

Li, Yafang, and Yisong Li. 2018. "E-commerce order batching algorithm based on association rule mining in the era of big data." In *2018 Chinese Control And Decision Conference (CCDC).* IEEE, June. doi:`10.1109/ccdc.2018.8407443`.

Liu, Peng, Chen Zhou, Yaohua Wu, and Na Xu. 2008. "Slotting the Complex Automated Picking System in tobacco distribution center." In *2008 IEEE International Conference on Automation and Logistics.* IEEE, September. doi:`10.1109/ical.2008.4636515`.

Lundgren, Jan., Mikael. Rönnqvist, and Peter. Värbrand. 2010. *Optimeringslära.* Studentlitteratur.

McKinney, Wes. 2010. "Data Structures for Statistical Computing in Python." *Proceedings of the 9th Python in Science Conference.*

Meurer, et. al. 2017. "SymPy: symbolic computing in Python." *PeerJ Computer Science* 3 (January): e103. ISSN: 2376-5992. doi:`10.7717/peerj-cs.103`. `https://doi.org/10.7717/peerj-cs.103`.

Negahban, Ashkan, and Jeffrey S. Smith. 2014. "Simulation for manufacturing system design and operation: Literature review and analysis." *Journal of Manufacturing Systems* 33, no. 2 (April): 241–261. doi:`10.1016/j.jmsy.2013.12.007`.

Nils Boysen, Felix Weidinger, René de Koster. 2018. "Warehousing in the e-commerce era: A Survey." *European Journal of Operational Reserach.*

Oskarsson, Björn, Håkan Aronsson, and Bengt Ekdahl. 2013. *Moderrn Logistik.* Liber.

Pazour, Jennifer A., and Russell D. Meller. 2011. "An analytical model for A-frame system design." *IIE Transactions.*

Raschka, Sebastian. 2018. "MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack." *The Journal of Open Source Software* 3, no. 24 (April). doi:10.21105/joss.00638. http://joss.theoj.org/papers/10.21105/joss.00638.

Robinson, Stewart. 2014. *Simulation: The Practice of Model Development and Use, 2nd edition.* Red Globe Press, April. ISBN: 978-1137328021.

Sargent, R G. 2013. "Verification and validation of simulation models." *Journal of Simulation* 7, no. 1 (February): 12–24. doi:10.1057/jos.2012.20.

SciPy. 2020. "scipy.stats.normaltest." https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.normaltest.html.

SSI Schäfer. 2019. "Apotea automatiserar med SSI SCHÄFER" [in Swedish]. April. https://www.ssi-schaefer.com/sv-se/om-oss/senaste-nytt/apotea-automatiserar-med-ssi-schaefer-567688.

A Technical Specifications

To increase the replicablity of the study the technical specifications of the hardware and the software used in the study are presented in this appendix. As reported by Gallagher (2019), some scientific studies were affected by a glitch in the Python language that occured depending on the operating system used. This is a further reason to include the technical specifications

A.1 Hardware

These are the specifications for the computer used in the study:

- **CPU:** Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz
- **RAM:** 16GB DDR4 2133 MHz
- **Operating System:** Ubuntu 19.10

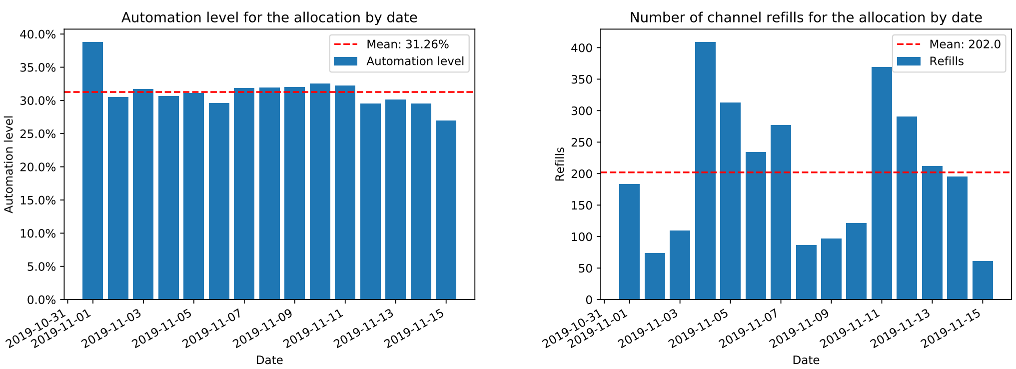
A.2 Software

These are the main software packages used in the study:

- **Programming language:** Python 3.7.5
- **Data Analysis:** pandas 0.25.3
- **Plotting:** matplotlib 3.1.1
- **Association rule learning:** mlxtend 0.17.0
- **Simulation Environment:** simpy 3.0.11

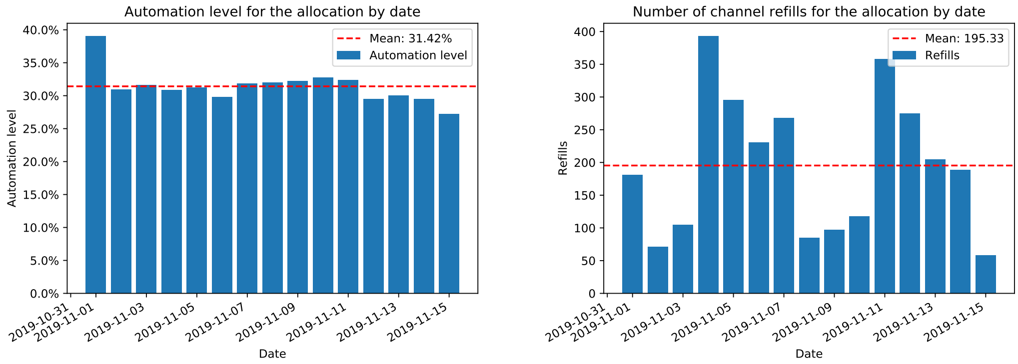
B Heuristic with other minimum supports

This appendix presents the other tested minimum supports for the heuristic algorithm. The tested minimum supports were 0.0000025, 0.00002 and 0.0001



(a) The automation level (b) The number of total channel refills

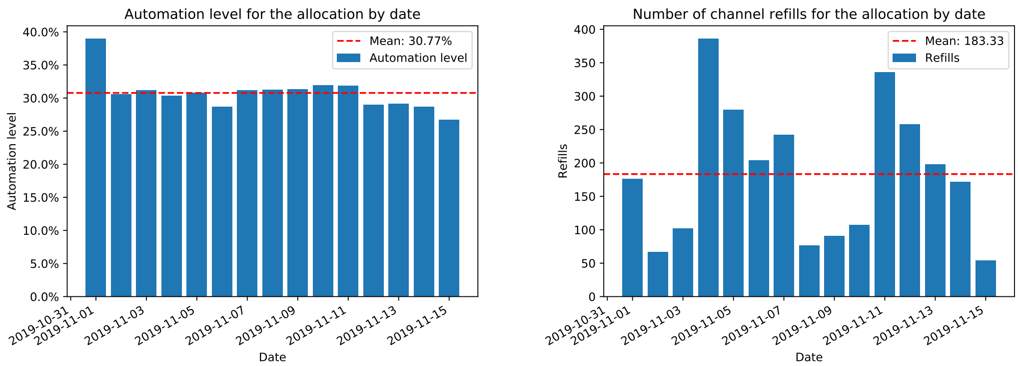
Figure B.1: Allocation by heuristic minimum support = 0.0000025



(a) The automation level

(b) The number of total channel refills

Figure B.2: Allocation by heuristic minimum support = 0.00002



(a) The automation level

(b) The number of total channel refills

Figure B.3: Allocation by heuristic minimum support = 0.0001

C Developed code

All the code used in the study can be seen at `https://github.com/aleda145/product_allocation_aframe`

The file structure is briefly explained.

C.1 Allocation

C.1.1 generate_selection_from_quant_only_all_skus.py

Generates an allocation using all orders.

C.1.2 generate_selection_from_quant_only_ok_skus.py

Generates an allocation using only valid orders.

C.1.3 greedy_graph_search.py

Places the association rules in a graph structure, then solves it using a heuristic.

C.1.4 change_support.py

Changes the minimum support for the generated rules.

C.1.5 draw_graph.py

Draws the generated association rules as a graph structure.

C.2 Data Analysis

C.2.1 orders.py

Takes in raw data and generates the training and testing data sets used for the rest of the thesis.

C.2.2 orders_for_date.py

Splits the data generated by orders.py into days to be used in the simulation.

C.2.3 sku_in_channel.py

Finds how many products can fit in one channel.

C.2.4 article_dimensions.py

Takes a generated aframe allocation and merges the information from the sku_in_channel.py to get a data file that can be used in the simulation.

C.2.5 find_max_automation_level.py

Calculates the maximum automation level assuming the A-frame has an infinite capacity.

C.2.6 count_sku_support_all_skus.py

Counts the support for the products that can fit in the A-frame using all orders.

C.2.7 count_skus_support.py

Counts the support for the products that can fit in the A-frame using only valid orders.

C.2.8 count_sku_quantity_per_day.py

Counts which products overflowed their channel.

C.2.9 test_normality_for_skus.py

Performs a normality test on each product.

C.2.10 sku_in_channel.py

Assigns the number of channels for a product using either the mean demand or the median demand.

C.2.11 aframe_over_time.py

Plots the resulting data from the simulation to different figures.

C.3 Machine Learning

C.3.1 association_rules.py

Performs the FP-growth algorithm on the data set.

C.3.2 clean_rules.py

Cleans the rules generated by FP-growth.

C.4 Simulation

C.4.1 aframesimulation.py

This is the simulation class. It is a big file and is heavily documented. Please read the documentation in the file.

C.4.2 aframe_sim_by_day.py

This performs a simulation of each day examined in the study.