Hybrid warehouse optimisation

Improving warehouse performance through buffer allocation and manual picking configuration

EPA2942: Master Thesis

Timo Frazer

Delft University of Technology

TU Delft

Hybrid warehouse optimisation

Improving warehouse performance through buffer allocation and manual picking configuration

by

Timo Frazer
4579992

Chair & Supervisor:	Prof.dr.ir. A. Verbraeck
Supervisor:	Dr. S. Fazi
Supervisor Picnic:	W. van Eeghen
Faculty:	Faculty of Technology Policy and Management, Delft

Cover: Picnic Technologies
Style: TU Delft Report Style

Executive summary

This research focuses on improving the performance of a hybrid warehouse system. A hybrid warehouse combines automation with manual processes. Both processes are combined with a buffer to catch up with the differences in these processes. The buffer has to be big enough to catch up with these disturbances, and smart allocation strategies can help efficiently use the buffer. If the buffer overflows, the process preceding the buffer will be affected, decreasing the warehouse's performance.

This study utilizes a case study of an online grocer, picnic. The case study system is a new hybrid fulfilment centre in Dordrecht. This case study system has a buffer with 20 parallel conveyor buffer lanes. Orders entering the system must be allocated to a buffer lane for this parallel buffering. This allows for different allocation strategies. The automated system and buffer are followed by a manual picking (cart pick) system in the case study system. The fact that the cart pick groups orders together allows for different grouping strategies, which can influence the state of the buffer as well as the productivity of the workers. Also, the configuration of the manual process will impact the system's performance.

For this research, the following research question has been established:

> *'What set of strategies for the input buffer and the manual process is most effective for improving the performance of a hybrid warehouse system?'*

This research question will be answered with the help of the following six sub-questions:

1. What are the performance metrics of hybrid warehouse systems?
2. What are the strategies for the buffer?
3. What are the possible configurations of the manual process?
4. What is the influence of strategies for the buffer on the performance metrics?
5. What is the influence of different configurations for the manual process on the performance metrics?
6. What are the effects of disturbances on the performance metrics?

The first three sub-questions are answered by analysing the system and by talking to experts within Picnic. The most important performance metrics are:

- Timeliness of orders
- Sojourn time of orders
- Throughput time of stack
- Productivity of pickers

The system is translated into a simulation model where the buffer strategies can be varied, and the configuration of the manual process can be determined. Through comprehensive experimentation, the study reveals that a priority lane is an element that should be implemented. With a priority lane, orders that are close to their deadline when they enter the system are put into the priority lane. When a picker requests a group for a pick round, the priority orders are prioritised and will always be picked first. The number of late totes decreases by 83%, as seen from Figure 1. For the strategies, prioritizing the deadline of totes for buffer lane selection and the Inter Process Buffer (IPB) time for group formation results in the best system performance, which can be seen from Figure 2.

From the results, it is also be suggested that the configuration of the manual process should consider the group size to be adjusted based on the expected workload of the day. The number of late orders increases as the minimum group size increases, while the picker productivity increases as the minimum group size increases.

For the results regarding the disruptions, it is concluded that if a consolidation station breaks on a peak day, the station should be repaired within 90 to 120 minutes if there are enough pickers and between 60 to 90 minutes if there are just enough pickers. This is also shown in Figure 3. For this

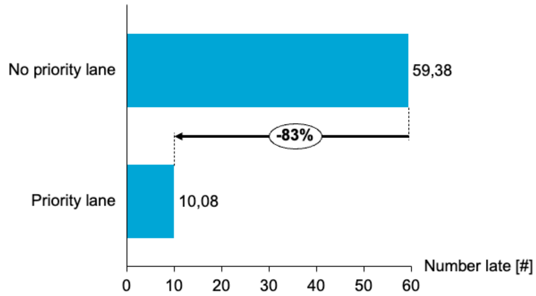

Figure 1: Results priority lane

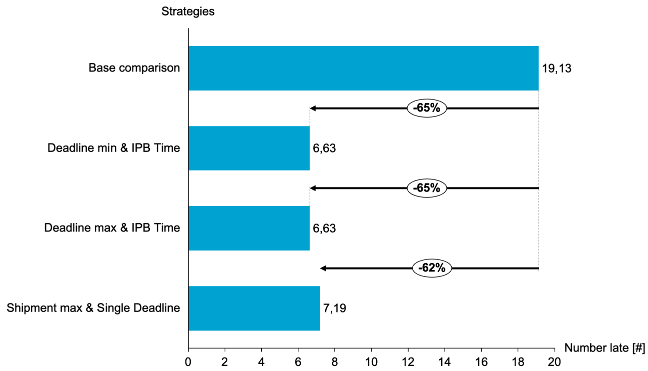

Figure 2: Results strategies

experiment, it is important that the buffer would not overflow. If the buffer overflows, it will impact the automated process preceding the manual process, significantly impacting the whole warehouse's performance. As the case study system has a capacity of 22 orders per lane, the maximum number of items in a buffer lane should not exceed 22. As can be seen, this happens between 90 to 120 minutes.

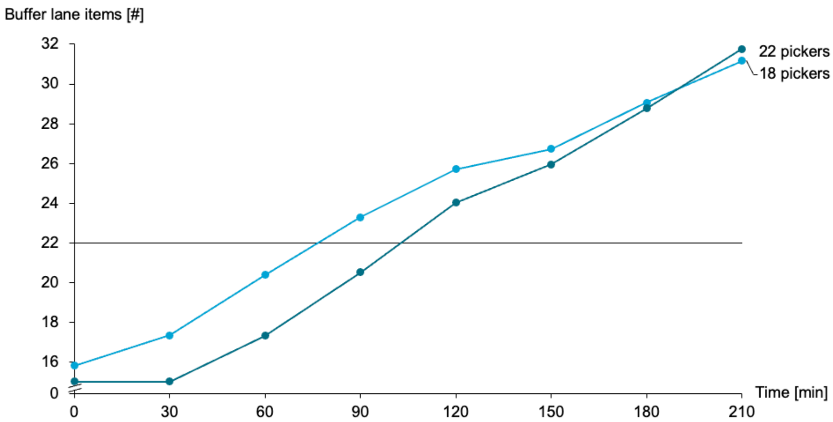

Figure 3: Results consolidation outage

After analysing the case study results, the findings can be generalised to apply to a broad range of hybrid warehouse systems. The first important thing to note is that it should always be investigated how easy it is to buffer. The case study system has a big buffer with many parallel conveyor buffer lanes. This simplifies buffering and allows for easy implementation of intelligent buffer allocation strategies. These strategies can lower the buffer fill rate and control the system's output. If possible, for buffer lane allocation, prioritise the item's deadline.

Another thing that can be concluded is to create a separate process for urgent items. This separate stream for urgent items allows those items to be handled outside of the buffer system and reduces their processing time, increasing the performance of the warehouse.

It is advised to identify physical capacity blockers if disruptions occur when the system operates at capacity. Physical capacity blockers are parts of the system that can not be scaled up, like adding more human workers to mitigate the effect of losing one worker. These physical blockers should be repaired or solved in time when the system is operating at capacity. This also means that maintenance operators should be available during these peak days.

In conclusion, this research offers valuable insights into optimizing hybrid warehouse systems' performance, with practical implications for buffer allocation, urgent item handling, and capacity management. The findings can be applied to various hybrid warehouse settings to enhance operational performance.

Contents

List of Figures

List of Tables

Glossary

Table 1: Glossary

Term	Definition
Key Performance Indicator (KPI)	Important indicator for measuring the performance of a system.
(Order) Tote	Plastic crate which moves along the conveyor and to which the products are added to assemble the order.
Inter Process Buffer (IPB)	Buffer between the two picking methods in the hybrid warehouse where order totes are stored.
Cart pick	Manual picking technique where a worker pushes a cart with order totes past shelves and collects the products.
Pick cart	Cart which the worker pushes with order totes that need certain products.
Zone pick	Automated picking technique where the order totes come to a worker. The worker has a specific zone with articles and puts the desired article in the tote.
Stock keeping unit (SKU)	Used to express one specific article.
Orderline	Used to express one specific article in an order.
Discrete Event Simulation (DES)	Simulation technique that models the system as a series of events that occur at specific points in time.
System Dynamics (SD)	Tool to model and analyse the dynamic behaviour of complex systems over time.
Agent-Based-Modelling (ABM)	Simulation technique used to model the behaviour of complex systems by representing individual entities (agents) and their interactions within an environment.
FCA	Picnic's automated fulfilment centre.
Goods To Person (GTP)	A picking strategy where the stock comes towards the person picking the goods.

1 Introduction

This research will look into improving hybrid warehouse performance through a simulation model. Section 1.1 will highlight this research's problem. It explains how the problem arose and what has already been done in warehouse research. Section 1.2 explains the case study that will be done to research the problem. Section 1.3 highlights the main research question and the supporting sub-questions. Section 1.4 briefly highlights the methods that will can be used for this research. Finally, Section 1.5 outlines the method selection for this research.

1.1. Problem introduction

Automation is increasingly applied in warehouses and distribution centres (Azadeh et al., 2019). Where robotised handling and automation used to be expensive and required a lot of space, technological advancements have improved warehouse automation and made it a feasible solution across a wider range of applications. Automation is taking over human processes in warehouses. Humans are an important aspect of warehouses, but humans also cause uncertainty in the output of warehouses (Cimini et al., 2019). Automating a warehouse will make the warehouse more predictable. However, when a warehouse needs to handle many different kinds of products, robots have a hard time handling the products. Also, delicate products like fruit and vegetables are still best handled by humans. Humans are more versatile and more 'intelligent' regarding their capacity to react to different situations (Bruzzone et al., 2007).

The high speed and large volume of automation and the versatility of humans can also be combined in a warehouse. A warehouse which combines automation and manual processes will be called a hyrbid warehouse in this research. A hybrid warehouse allows for different combinations which optimally use the advantages of both automation and humans.

This research will focus on the challenges that arise in hybrid warehouses. One challenge is the difference between the two processes, which must be captured. If, for example, an order moves from an automated part of a system to a manual part of a system, there will be differences and fluctuations in the processing times and capacities of both processes. The buffer between the processes should be able to catch up with the differences in those processes. If the buffer is not big enough, this will cause problems in the first process, slowing down the whole operation. Something that increases the complexity of this challenge is that humans have an important role in hybrid warehouses. The buffer should be able to catch up with disturbances and variability in human performance.

1.2. Case study

For this research, a case study will be conducted. This research focuses on a warehouse of online grocer, Picnic. Picnic was founded in the Netherlands and operates in Germany and France. Picnic's manual picking centres have been improving a lot over the years. Still, to be able to handle much larger volumes, they opened a highly automated fulfilment centre in Utrecht in 2022. This highly automated fulfilment centre has shown that automation can benefit the company.

Picnic's hybrid warehouse combines two types of picking strategies. One strategy uses automation where the order tote (crate in which the order is assembled) comes up to a station with a worker.

The station contains only a certain amount of products to eliminate/minimise the worker's travel time. The order tote moves on a conveyor along all stations that contain products needed in the order. This picking strategy is called zone pick. After the picks from zone pick of an order have been done, it moves along to the manual picking area. In this area, the products are picked using the cart pick strategy. A worker walks along shelves with products and adds items for an order to a compartment on the cart. These two processes are split into fast movers and slow movers. Fast movers are products that are ordered more often and are picked with the automated solution. While slow movers are products that are ordered less often and are picked using the manual picking strategy. As this warehouse combines these two strategies, one being highly automated and one being a manual picking strategy, the name hybrid warehouse was chosen.

The case study system that will be studied in this research has multiple parallel buffer lanes. The order totes that enter the system must be allocated to a buffer lane based on different order properties. The allocation strategies used to allocate the totes to the buffer lanes will impact different KPIs. Understanding how the different allocation strategies impact the system's performance is important because if the system operates at capacity and the buffer is not used efficiently, the buffer will overflow. If the buffer overflows, dieback will cause the zone pick system preceding the manual pick to stop, eventually halting the whole operation until the manual pick has recovered. Also, not using a smart strategy to fill the buffer lanes will most likely cause more totes to be late as totes will not be grouped based on the deadline, so totes will always have to get in the back of a longer line and do not have a shorter line to be picked from faster. Also, the strategies are expected to impact human performance and allow the buffer to better catch up with the variability of human actions. Using different strategies it is expected that the human workers can have increased productivity, and smart strategies will impact how the buffer is used to catch up with the variability of the human actions.

It is expected that smart strategies will improve the whole system's performance. The state of the buffer, the system's output, and the human workers' productivity are all important indicators to measure the performance of the warehouse. This research will aim to optimise the buffer allocation and group formation strategies and the configuration of the manual picking section.

As said above, the research will be based on a case study at an existing warehouse. The warehouse is located in Dordrecht and is not yet in operation. The first orders will be fulfilled at the beginning of October 2023.

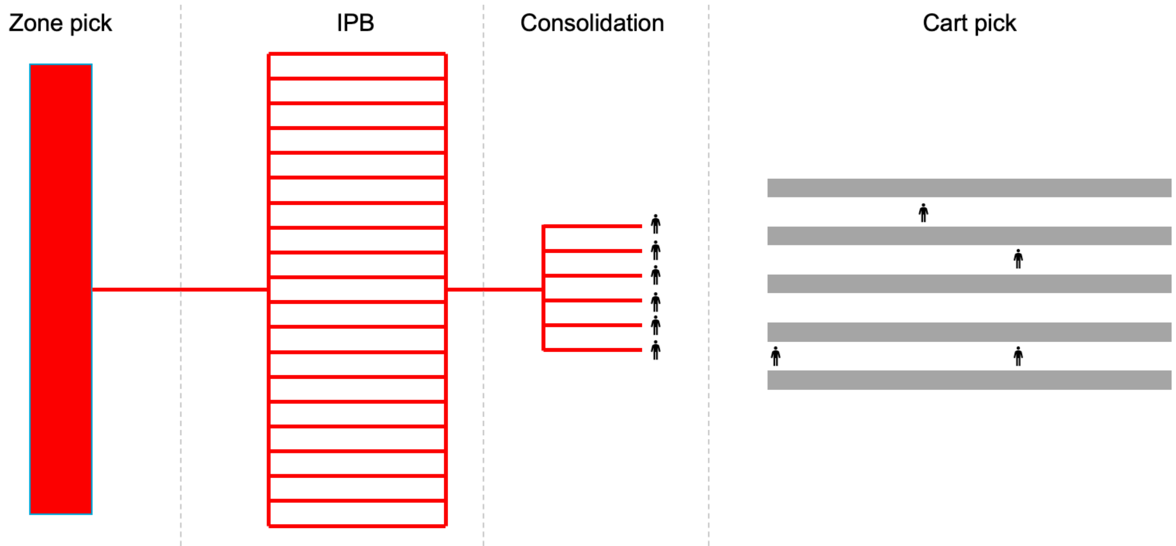

Figure 1.1: Highlighted scope

In Figure 1.1, a section of the full warehouse is shown. This is the section where the Inter-Process Buffer (IPB) and the cart picking circuit are located. The inter-process buffer- and cart-picking processes are highly important to this study. The model of this research will look at orders from the moment they enter the IPB until a worker is done with consolidating the order. This whole process can be split up into four parts which are extensively explained in Chapter 3. These four processes are:

1. Lane selection in the IPB.
2. Batch formation for cart picking.
3. Cart picking round.
4. Consolidation of the order.

These 4 processes are important in this research, and each will impact the system's performance. The first two involve the buffer, and the last two contain the human processes. These processes each have challenges and will be modelled with increasing complexity starting from a small/minimal model and building up to increasingly complex models.

1.3. Research question

The research question will comprise the two main components described, the buffer strategies and the configuration of the manual pick area. To find the best combinations of the allocation strategies and the configuration of the manual pick section, the following research question has been established:

> *'What set of strategies for the input buffer and the manual process is most effective for improving the performance of a hybrid warehouse system?'*

The research question will be answered by researching the case study system, and results from this case study will also give generic findings which apply to other hybrid warehouses.

To aid in answering the research question, the following sub-questions have been established:

1. What are the performance metrics of hybrid warehouse systems?
2. What are the possible strategies for the input buffer?
3. What are the possible configurations of the manual process?
4. What is the influence of strategies for the buffer on the performance metrics?
5. What is the influence of different configurations for the manual process on the performance metrics?
6. What are the effects of disturbances on the performance metrics?

The first sub-question will determine the problem's performance metrics and what needs improvement. The second question will find the parameters impacting the system's performance. The third question focuses on the buffer at the beginning of the process, what strategies should be used to choose the lane in the buffer, and how to approach group formation. The fourth question will show the effects of the variance in the manual picking process on the system and what strategies can be used with worker assignment. The last question will focus on system elements that limit improvements and how they can be tweaked to improve the system's performance. The answers to the sub-questions together will form a conclusive answer to the main research question.

1.4. Research methods

Discrete Event Simulation (DES) is a simulation technique that can process many more events and interactions than other simulation techniques. In DES simulation, time skips to the next event on the future event list, which lists all events and when they will happen. So with DES, all the time between events does not have to be modelled, greatly decreasing the computational intensity of models.

Another research method that is often used for the buffer allocation problem is mathematical models/optimisation algorithms. Optimisation algorithms take an objective function that needs to be minimised or maximised. The solution is subject to different constraints which delineate the solution space. The optimisation algorithms can then be run to find the optimal solution to the problem.

Another modelling method that is often used is System Dynamics (SD). SD uses feedback loops, in and outflow and integral calculations to show the flow of variables over time. With these characteristics, SD is unsuitable for this research question and any sub-questions.

Another modelling method that could be interesting for the problem is Agent-Based Modelling (ABM). ABM models focus heavily on the behaviour of agents and the interactions between them. In a hybrid warehouse, there are also humans picking the products. Hence, ABM could be an interesting method to investigate. For example, ABM can be used to investigate how the picking process by humans can be optimised. In literature, some studies have already used ABM to model manual picking in warehouses (Rubrico et al., 2006). However, for this research, the human picking process will not be taken into the scope of this MSc thesis.

The system information used in this research will be from an existing hybrid warehouse in Dordrecht, The Netherlands. The hybrid warehouse is a warehouse of online grocer Picnic. Picnic is a very data-driven company. They have a special so-called ”data warehouse” in which they conveniently store all their data so analysts and researchers can easily access data from all their operations. This data will aid in answering the research question and sub-questions and creating the model.

The data should be analysed in detail to answer the first and second sub-question. Moreover, the hybrid warehouse in Dordrecht has been visited multiple times to see all the processes in action as this warehouse is already built. After the research, the findings can also be implemented into the warehouse software. In addition to visiting the site, interviews with experts within Picnic will find the most important performance metrics and what parameters can be changed to impact the outcome of the system.

To answer the third and fourth sub-questions, the outcomes of the DES model will be used. Using the simulation model, the parameters from the second question can be tweaked, and an analysis of the simulation outcomes through the performance metrics can be done to find which parameters have the biggest impact on the performance metrics and which are not of great importance. All data used in answering these sub-questions will come from the data warehouse of Picnic and information gathered from visiting the warehouse. Also, the strategies will be fed into the simulation to find the best strategies, and the performance metrics will tell what strategies work best.

To answer the fifth research question, the simulation model will be used. In the simulation, it can be easily identified where the bottlenecks of the system are. The bottlenecks can be identified and researched by looking at the queues, the utilisation rate and many more metrics. After the bottlenecks are found, their values can be tweaked to see their impact on the performance metrics.

With all the data gathered some extra data analysis will be done. The data analysis is done with Python using different packages Python. The data analysis will further aid in answering the research question.

1.5. Method selection

The possible approaches can be examined after describing the problem and breaking it into different parts. The two most present methods in the literature are simulation and mathematical models. For this research problem, mathematical models would be most applicable to the buffer. Lane selection into the buffer and group formation of orders from the buffer are subjects that can be formulated into objectives and constraints. These objectives and constraints can then be optimised using optimisation algorithms. A downside of these mathematical models is the randomness of the inflow of the orders is harder to use as input for a mathematical model. Another downside to mathematical models is that it might be hard to synthesize clear strategies from the output of the optimisation model, but this cannot be concluded yet.

Simulation will probably be the best method for the second part of the problem. The process of orders moving through the system, workers asking for groups and doing the cart pick round. Workers consolidating the order. All of these aspects, with their variances, can be put into the model. The model can then be run under different scenarios to examine the system's performance.

When looking at the problem, a combination of mathematical models and simulation can be a proper way to approach the research. Another approach could be only to use simulation. All the logic of lane selection and group formation in the buffer will be fitted into the simulation model. A downside when only using simulation would be that the strategies are determined beforehand. Thus, the ”optimal” strategy cannot be concluded.

Another option that could be looked into is to start with a simulation model of the system in its base state. Running this model over different scenarios could point out the important bottlenecks of the system. The output of this base model can then be used to determine what part of the process has the most influence on the system's performance, thus, what method to use to try and optimise the system's overall performance.

For this research, it has been decided that building a simulation model for the system would be the best option to fit the scope of the research. The simulation model could take the arriving totes, put them into a buffer lane based on different allocation strategies, and group them with grouping strategies. Variation and uncertainty can all be put into one simulation model, and the results will aid in answering the research question.

A limitation of the DES approach can be that DES models are modelled closely related to their real-world counterparts and can, thus, sometimes have a hard time finding new solutions. The modeller has to input new configurations himself. Thinking out many possible configurations can ensure that missing a possible good configuration combination is prevented.

1.6. Research structure

The research will be split into 8 chapters excluding this introduction. Chapter 2 will look into literature. What research has already been done into warehouses and hybrid warehouses? What research can aid in answering the research questions? Chapter 3 will give the case description. Sub-questions one, two and three will be answered in that chapter. Chapter 4 will explain how the system as described in Chapter 3 has been translated into a simulation model and Chapter 5 will validate that model. Chapter 6 will explain the experimental setup and Chapter 7 will discuss the results of these experiments. These two chapters will give an answer to sub-questions four, five and six. Chapter 8 will discuss the results and their implications for Picnic and general warehouses. Lastly, Chapter 9 will give the conclusion of this research and give a concluding answer to all the sub-questions and the research question.

2
Literature study

To assess the scientific relevance of the problem, a literature review is conducted. In the literature review, papers on this topic are reviewed, and a knowledge gap is identified. In Section 2.1 the search methodology is described. Hereafter, in Section 2.2 the early studies are described while Section 2.3 reviews recent studies. After which, Section 2.4 and Section 2.5 go more in-depth into the buffer allocation problem and modelling human behaviour. Finally, Section 2.6 gives an overview of the discussed literature.

2.1. Literature search methodology

During the literature review, Scopus is used to select relevant literature. The following queries are used to search for relevant literature.

- warehouse AND (queu* OR buffer)
- warehouse AND queu* AND (optimi?ation OR simulation OR mathematical)
- buffer AND allocation
- buffer AND allocation AND problem

As American English and British English use different spelling for the word optimisation, a question mark is used in the place of the letter s in that word to include all spellings.

For the recent work literature review, only papers after 2016 are selected. In addition to the query, the snowball method was used to find relevant articles based on the articles from the search results. for all results, highly cited papers were preferred above papers with few to no citations to select the best papers in the field. Papers from very different fields than logistics simulation and warehouse simulation were excluded from the review

2.2. Early studies

To select the literature of early studies, papers with the selected queries described in Section 2.1 which are published before 2016 are selected.

One of the first papers on this topic is by Bhaskaran and Malmborg (1989). It is a study into the effects of different types of picking strategies and their impact on the performance metrics of the warehouse. The paper investigates a storage and retrieval warehouse/system. Another paper and the first paper that mentions an automated warehouse with a storage and retrieval system is by Randhawa and Shroff (1995). This paper investigates different warehouse design and their impact on performance metrics. Pandit and Palekar (1993) also looked into warehouse design and its impact on performance. Other authors that have written multiple pioneering articles that have been cited in other papers are Eben-Chaime and Pliskin (1996). Eben-Chaime and Pliskin introduced new performance metrics and included queuing, which is important in the system that will be studied in this research. In their first article (Eben-Chaime & Pliskin, 1996), three different operating modes are investigated. In the second article (Eben-Chaime & Pliskin, 1997), a system with multiple machines is researched. From (Eben-Chaime & Pliskin, 1997), the optimal number of machines for the operations is determined.

Chew and Tang (1999) are the first researchers that use optimisation and simulation models to optimise the placement of products throughout the warehouse. Next to that, the amount of stock in the warehouse is optimised. Yu and de Koster (2009) determines how orders should be batched and how this impacts the warehouse's queues and other performance metrics. Yu and de Koster (2009) concludes that efficient order batching could greatly impact the system's overall performance. To further improve warehouse operations Roy et al. (2009) researched warehouse automation using autonomous vehicles. The research of Roy et al. (2009) also solved queuing problems in the system by determining the optimal amount of vehicles unloading the racks. The model in this research is validated by simulation. Pan and Wu (2009) is the first research with multiple pickers. In a warehouse, there will most certainly be more than one picker. Hence, there will be cases of congestion and other problems that come with the interaction between multiple pickers in warehouses. Pan and Wu (2009) also validate the research model with a discrete event simulation model and concludes that the model worked to improve the performance of the warehouse. An algorithm is developed by Naeem et al. (2010) for the first time using near-real-time data for warehouses. As discussed in Chapter 1, real-time data can be a very promising solution for warehouses as data from the current state of the warehouse is directly fed back into the simulation that is run.

For the queuing in warehouses Hulett and Damodaran (2014) are the first to do breakthrough research on fork and join queuing. The orders that come in are split among different pickers in the warehouse, and later the products form a new queue to be joined again into fulfilled orders. Around the same time, Balamurugan (2015) executes new research into how queues can be analysed. Balamurugan (2015) concludes with new optimisation algorithms and new evaluation methods for the status of a warehouse system.

Another interesting concept to analyse the system is the theory of constraints by (Goldratt, 1990) and reviewed by (Rahman, 1998). This theory states that every system has at least one constraint. By identifying the constraints and deciding how to exploit the constraints, the throughput can be maximised. Global decisions must be arranged to maximize the effectiveness of the most limiting constraint. Eventually, a new limiting constraint will emerge, and this circle can repeat itself. An application of the theory of constraints is critical chain scheduling (Herroelen & Leus, 2001). This theory analyses a process's 'critical chain' and determines the minimal/baseline schedule. Improving a process that is part of this critical chain immediately improves the system's overall performance. These researches are also important to look at when modelling a warehouse system in order to improve the right processes to improve performance.

As can be concluded from the early work. The first research in the field laid the groundwork for warehouse operations. Picking strategies and warehouse design/layout were the first aspects to be researched. From the early work, the most important performance metrics can be synthesised.

- Sojourn time of orders
- Total time in queue (of order)
- Time in stock (of product)
- Productivity of picker (number of orders)

The metrics listed above are the most important metrics to evaluate the performance of the warehouse. Next, a recent literature study will point out the research gap and the problem that will be addressed with this research.

2.3. Recent work

To find what more recent studies have done into warehouse optimisation recent work will be reviewed. Specifically the process of the buffer containing orders between two stations in a warehouse, the queues that form in this buffer, and how the system can be optimised by using different strategies to order this queue.

A very recent paper by Eder (2023) does research on how to merge items from different aisles into a limited queue/buffer under a fully sequenced order. Eder's warehouse uses a shuttle-based retrieval system, where shuttles collect the items from the aisle and bring the items to the conveyor that needs to put the items into the queue. This problem looks like the problem in this research where orders come in from automated picking isles and enter one of N queues. The order picker picks the last items after the queue reaches a particular length or if an order reaches its deadline. The paper by Eder has not yet researched splitting the orders into different aisles. Besides, Eder has only applied an analytical approach and has not verified the findings using simulation. Implementing the found strategies is also something that this research aims to do.

Malde et al. (2022) research different buffer streams in a manufacturing plant and how to optimise team formations for the system to perform effectively. For the problem in this research, we need to investigate how to optimise the buffer layout and how to add orders to buffer lanes. The article by Bansal and Roy (2021) researches an integrated storage-order picking system. In this system, the items move from the storage to the picker automatically, and the picker finishes the order. The research of Bansal and Roy (2021) aims to optimise the upstream and downstream networks. Hence, the storage and retrieval system to the order picker and his system. Moreover, validation is done using a simulation. Results from Bansal and Roy (2021) can be used in the research of this paper to test if different upstream strategies impact the performance of the buffer to the order pickers and the whole system.

Another interesting research uses a technique mentioned in Chapter 1. Shang et al. (2022) set up a method to get real-time information on the warehouse and the queue using a digital twin. As explained before, a digital twin is an exact copy of the warehouse or part of the warehouse in a simulation. In this research, information from the warehouse is fed into the simulation and the simulation can anticipate what actions can best be taken in the warehouse to improve operations.

2.4. Buffer allocation problem

The buffer allocation problem is a specific part of the literature that looks similar to this research problem. The term first appeared in a paper by Sarker (1984). A lot of different studies have been done on the buffer allocation problem. Most research on the buffer allocation problem can be split into two methods. Analytical optimisation methods and simulation (-optimisation) methods (Smith & Cruz, 2005). Where Shaaban and Romero-Silva (2021) and Smith and Cruz (2005) and other papers have also stated that simulation methods are mostly slower for smaller systems but excel when dealing with larger class systems compared to analytical methods. Kose and Kilincci (2020) studies the optimal buffer size using multi-objective optimisation. Mixed integer linear programming is also used by some studies, for example, by Magnanini et al. (2022). A recent study that uses discrete event simulation is by Shaaban and Romero-Silva (2021). Using a model created in Simio, two merging lines with uneven buffer capacities are optimised.

2.5. Modelling human behaviour

Something that cannot be overlooked is the fact that humans still play a part in warehouse processes (Cimini et al., 2019; Dewa et al., 2017). To make the human as predictable as possible, operations are getting close to make a human as predictable as a robot by letting computers decide everything. Humans will do whatever the computer tells them to do (Winkelhaus et al., 2021). This is even starting to become part of everyday life. Think about computer assistance when driving a car. The latest cars can take driving completely out of the hands of the human driver. Computer assistance is also increasingly applied in warehouses, but humans are still the superior rational decision-makers. They are 'intelligent' in terms of their capacity to react to different situations (Bruzzone et al., 2007). This, together with the fact that humans are still superior in picking many different kinds of objects, is something that heterogeneous automated picking is trying to solve, (Grambo et al., 2019). So, the fact that humans are still more 'intelligent' is why humans are still an important part of the system that will be studied in this research. There are about 10,000 different products in all shapes and sizes. Humans are still needed to handle these products.

So humans are still needed in warehouses. However, humans also cause fluctuation in output. Humans make errors (Dewa et al., 2017) and can behave differently than the computer expects them to. Human errors and unexpected behaviour make modelling the process's human aspect difficult. This variance must be considered when modelling human interaction processes to account for human behaviour. The buffer in the system to be researched also allows for this variance and is a key element in enabling the human element in the process.

2.6. Literature overview

Much general research has been done on warehouse optimisation in the early literature. As warehouses grow in size and efficiently running operations become more and more important, this field of study expanded as more researchers started to research this. As automation becomes a bigger part of warehouses and more big warehouses implemented automation, this field of study also grew, and more research was done. With these developments, new challenges emerged. Buffers are used to catch up with differences between processes and ensure that each part of the system can operate at its own pace. Many studies have been done to optimise the length of the buffer, aiming to keep the buffer as small as possible while ensuring a certain performance. Research aimed at parallel buffers, like Eder (2023) and Malde et al. (2022), can help in this research as this study also works with parallel buffer lanes.

As different studies have researched how to optimise buffer length and sequencing. A gap in the literature and other research emerges. The problem described in Chapter 1 aims to optimise order allocation to buffer lanes based on different order properties. These order properties have an impact on the KPIs as well as the total throughput of the system and average sojourn time. A good balance between strategies for maximum throughput and lane selection based on order properties is needed to achieve optimal performance. This balance is not yet specifically researched in literature and, thus, this research can add to existing literature.

The last aspect to consider in this research is the human element in the system. Human behaviour is unpredictable, causing disturbances and variance in the system. As humans are still an important aspect of warehouse operations, the human aspect in warehouses has been broadly studied. These studies can also aid in investigating the human impact on the system. The buffer and the strategies that are used must catch up with disturbances and variances of the human element.

For the buffer allocation problem Table 2.1 shows a separate overview of the literature and used methods on the Buffer Allocation Problem. Table 2.2 shows all the literature and their concepts that are discussed in the rest of the literature review.

Table 2.1: Buffer Allocation Problem Literature overview

Author(s)	**Year**	**Method**	**Concept**
Sarker	1984	Literature overview	First mention of the Buffer Allocation Problem (BAP)
Smith and Cruz	2005	Optimisation approximation and simulation	parallel line BAP
Kose and Kilincci	2020	Multi-objective evolutionary optimisation	Single production line BAP
Shaaban and Romero-Silva	2021	Discrete Event Simulation (DES) model in Simio	Two parallel lines BAP
Magnanini et al.	2022	Mixed Integer Linear Programming	Single line BAP optimisation

Table 2.2: Literature overview

Author(s)	Title	Year	Concepts
Bhaskaran and Malmborg	Modelling the service process in a multi-address warehousing system	1989	picking strategies
Goldratt	Theory of constraints	1990	Theory of constraints
Pandit and Palekar	Response time considerations for optimal warehouse layout design.	1993	Warehouse design
Randhawa and Shroff	Simulation-based design evaluation of unit load automated storage/retrieval systems.	1995	Warehouse design
Eben-Chaime and Pliskin	An integrative model for automatic warehousing systems.	1996	queue length
Eben-Chaime and Pliskin	Operations management of multiple machine automatic warehousing systems.	1997	multiple machines
Rahman	Theory of constraints: A review of the philosophy and its applications	1998	Theory of constraints
Chew and Tang	Travel time analysis for general item location assignment in a rectangular warehouse.	1999	Item location
Herroelen and Leus	On the merits and pitfalls of critical chain scheduling	2001	Critical chain scheduling
Bruzzone et al.	Evaluation of the impact of different human factor models on industrial and business processes	2007	Human intelligence cause better decision making in warehouses
Yu and de Koster	The impact of order batching and picking area zoning on order picking system performance.	2009	Order batching
Roy et al.	Impact of zones on throughput and cycle times in warehouses with autonomous vehicles.	2009	automation using autonomous vehicles
Pan and Wu	Throughput estimation in a picker-to-part warehouse with multiple pickers and blocking considerations.	2009	Multiple pickers
Naeem et al.	R-meshjoin for near-real-time data warehousing.	2010	Real-time simulation in warehouses
Hulett and Damodaran	Fork and join queuing networks to predict order picking times at a warehouse.	2014	Fork and join queuing in warehouses
Balamurugan	The operation data analysis research based on the queuing theory of warehouse optimization.	2015	Queuing optimisation and warehouse evaluation methods
Dewa et al.	Human Errors in Warehouse Operations: an improvement model	2017	Human error in warehouse modelling
Cimini et al.	Exploring human factors in Logistics 4.0: empirical evidence from a case study	2019	Humans in warehouses
Grambo et al.	Automatic Sorting-and-Holding for Stacking Heterogeneous Packages in Logistic Hubs	2019	Heterogeneous product picking
Bansal and Roy	Stochastic modeling of multiline orders in integrated storage-order picking system.	2021	Items to and from order picker station
Winkelhaus et al.	Towards a conceptualisation of Order Picking 4.0	2021	Making human predictable (robotise)
Malde et al.	Optimal team formation and job assignment to optimize warehouse operations	2022	Different buffer streams
Shang et al.	Design of cross-platform information retrieval system of library based on digital twins.	2022	digital twin and queuing information
Eder	Analytical approach of merging a different number of storage aisle under a fully sequenced order.	2023	Merging items from different aisles in sequential order.

3

Case description

In this chapter, the case description will be given. It dives deeper into the problem that will be tackled in this research and what the layout of the Picnic fulfilment centre looks like. The chapter will first explain the layout of the fulfilment centre in Dordrecht, after which it will explain the problems to be tackled per section of the system. Finally, Section 3.2 will explain the important performance indicators for the case study.

3.1. System description

In the hybrid fulfilment centre in Dordrecht, an order tote will first go through the zone pick section of the warehouse. If all zone pick picks for an order are done, it will move on to the manual pick section. As explained in Section 1.1, this is where the challenge for this research is. The automated zone pick section has to be connected to the manual, cart pick, section. To compensate for the difference between these processes a buffer is implemented. The allocation strategies of totes in the buffer and the configuration of the cart pick section will have to be determined. This section will go through all the aspects of the problem and explain the challenges.

A buffer is needed between the two pick processes to accommodate the different processes. The buffer consists of 20 parallel buffer lanes. Because of this, lane selection is very important. The lane selection will be explained in subsection 3.1.1. Totes are stored in the buffer, and pickers do a pick round for a group of totes. Pickers will walk a pick round through the cart pick circuit as shown in Figure 3.1. A picker will collect the products of an order on a pick cart. When a picker scans the pick cart, a group will have to be selected if a group is available in the buffer. The group is formed using a group formation strategy. Group formation depends on lane selection, as totes allocated to a specific buffer lane will stay in that buffer lane in that order. The group formation is further described in subsection 3.1.1. While the picker is doing the picking round, the totes stay in the buffer lane, which adds to the importance of good lane selection and grouping strategies. Totes in a buffer lane can block other totes in that lane if they are behind and in another group.

Once the picking round is done, the pick cart is put into a pick cart queue and a consolidation worker can scan the pick cart. When the pick cart is scanned, the order totes for that group will move from the IPB to the consolidation station. The totes will move along a conveyor and end up at the consolidation station. At the consolidation station, the worker will put the products for an order at the consolidation position from the specific pick cart slot into the order tote. After the consolidation is done, the tote will exit the consolidation station along a conveyor and move on to dispatching before it is loaded into a stack and can be transported by truck to a hub to be delivered to the customer. Trucks must leave at a specific time, giving totes a deadline.

3.1.1. Buffer strategies

Different buffer strategies are possible in the system. A buffer lane can be used as a priority lane which is expected to improve timeliness. for the buffer lane allocation and group formation different strategies can be used. These buffer strategies will be explained in the following sections.

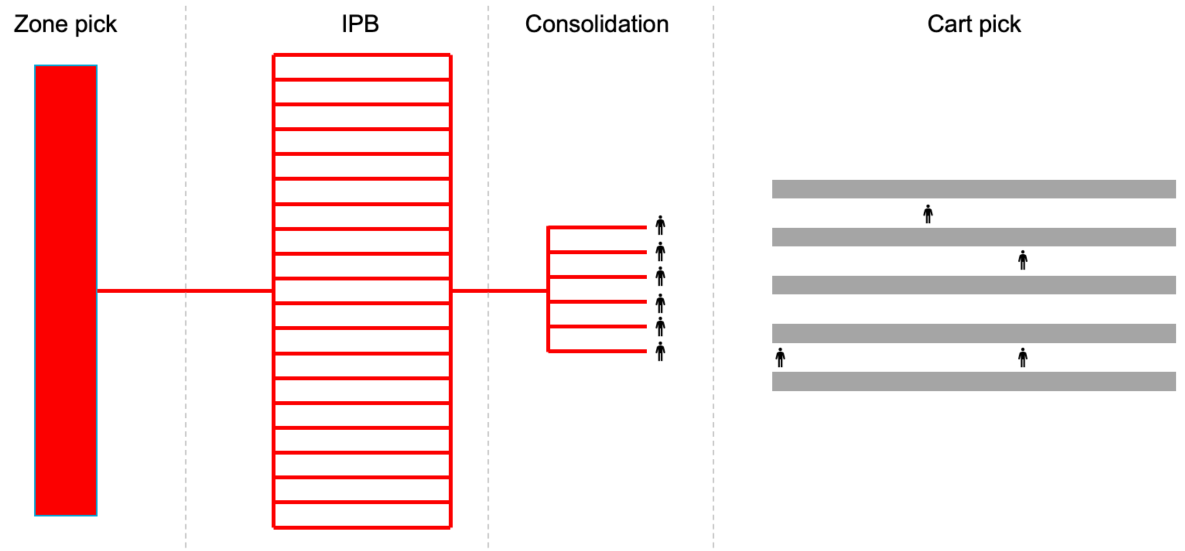

Figure 3.1: Highlighted scope

Priority lane

A priority lane is one of the options that can be implemented in the system. If a priority lane is used, one of the buffer lanes will be appointed as a priority lane. This means there are only 19 'normal' buffer lanes decreasing the buffer capacity for normal procedures. A priority lane allows for totes that enter the system close to their deadline to enter the priority lane. When a picker scans a pick cart to request a group from the buffer, all totes in the priority lane will be offered as a group. This means that pickers will sometimes start a pick round for a very small group, so their productivity will decrease. This priority lane allows totes close to their deadline to be processed faster, but others waiting in the IPB will have to wait longer. This priority lane will thus decrease the capacity of the buffer. It will decrease the productivity of workers. Also, it will increase the sojourn time of other totes. Part of this research will find out if a priority lane improves the system's performance.

Lane selection

A lane is selected based on the order properties and the lane properties. The most important aspect to consider is the deadline for an order. Totes can also take longer in the process that precedes this system and enter the buffer needing fast service. Grouping orders that end up in the same truck could be good for system performance because it allows the order to leave the buffer after the studied system faster. So it might not be a good idea to spread them across lanes/groups. Another order property that could impact system performance is the number of products the order still needs. The amount of products that need to be picked directly impacts how long the cart-picking round will take, as well as the consolidation of the order. Next to that, the type of products could be considered when choosing the buffer lane. These properties are not yet all properties that can be taken into account. Reviews at current sites and the data could show more properties that could be used in the research.

The buffer properties are also important to consider when choosing a lane. The number of orders already in a lane and whether they are already assigned to a group (meaning a picking round has already started) is important. In subsection 3.1.1, the group formation is explained further, but orders that are not assigned to a group yet can block other orders behind it. Also, because of the way group formation impacts the lane selection process, the number of empty lanes could be an important property. The buffer in the case study consists of 20 lanes with 22 spaces for totes.

Lane selection depends on group formation, and group formation depends on how the lane selection is done. This allows for different configurations of both processes, impacting the overall performance. A good combination of strategies will probably lead to a better result

After talking with experts within Picnic and analysing the system, properties to use within the strategies were determined. The properties that could be included are:

- Number of totes in lane

- Minimum group size
- Maximum group size
- Shipment ID of tote
- Stack ID of tote
- Deadline of tote

The following lane selection strategies have been formed:

- Base strategy
- minimum group size first
- minimum group size first - fill lanes one by one
- Maximum group size first
- Prioritise shipment - minimum groups first
- Prioritise Shipment - maximum groups
- Prioritise stack - minimum groups first
- Prioritise stack - maximum groups
- Prioritise Deadline - minimum groups first
- Prioritise Deadline - maximum groups

Appendix B shows the flowcharts of the strategies determining the lane a tote will go in.

Group formation

Group formation is another important aspect of the problem. As seen in subsection 3.1.1, group formation depends on how the orders are aligned in the buffer. Group formation decides for which orders a cart pick round is started. The algorithm could also consider the order's product properties in this group formation. The product properties could impact the time it takes to pick products in the cart pick round. Picnic will eventually use an algorithm to do the group formation. Once a picker scans a pick cart, an algorithm will decide if a group will be formed. Moreover, the algorithm will decide how that group will be formed and which group will be assigned to the picker. Therefore, this limits the algorithm's complexity, as it is not feasible to let an algorithm run for 5 minutes before assigning a group to the picker.

For group formation, it is also important to see the impact of group size on the system's performance. Bigger groups mean the cart pick round will be longer, which means more orders in the buffer will be reserved for longer. Causing the buffer to be filled more overall. On the other side, picking for bigger groups is expected to have a big positive impact on the productivity of the manual picking process, including consolidation. These hypotheses and their consequences need to be tested using the model. For the group formation, the following strategies have been determined:

- Base strategy
- Minimum IPB time
- Closest total deadline
- Closest deadline of single tote

Appendix B shows the flowcharts of the strategies determining the group formation.

3.1.2. Manual picking

All the products for a formed group are picked in the cart picking section. When an employee (M) grabs an empty picking cart, it is scanned, and the group formation (see subsection 3.1.1) is done. The employee picks all products into pockets on the pick cart corresponding to a specific order and adds the cart to a consolidation station after the cart is completely picked. A cart consists of a certain amount of pockets, and a worker takes this cart along the route and picks the products as described above.

At the consolidation station, an employee (C) scans a full picking cart and takes it to the station. After scanning, the group in the buffer corresponding to that cart can be released to make its way to the consolidation station. Afterwards, the worker can add the products from the pockets in the cart to the order in front of the employee. The order is then fulfilled and can go to the dispatch, where they will continue their journey to the customer.

These last two processes include humans performing the actions, which causes variance in how the actions are performed. Not all humans perform tasks in the same or even the right manner. The human-machine interaction causes fluctuations in the system's behaviour, so this has to be modelled in a way. This human-machine interaction will impact the system's performance.

3.1.3. Disruptions

In manual picking disruptions in the system can influence the performance of the system. The most significant blocking disturbance identified is when one consolidation station is broken. One-sixth of the consolidation capacity is missing which likely significantly impacts performance on a peak day. One aim of this research is also to analyse how quickly a part of the system should have to be repaired when operating at capacity. The buffer will be able to catch up with this disturbance up to a certain point.

3.2. Performance indicators

Interviews with experts were conducted to find the most important performance metrics for Picnic. From these interviews, some additional performance indicators were found next to the performance indicators from the literature. The most important performance indicator for Picnic was the number of late orders. This will also be important for other warehouses. For Picnic, a late tote will impact the timeliness of a whole shipment. If an order tote is too late, it will get cancelled, which has a big customer impact. A lower sojourn time will mean that the system can handle more totes, and thus the capacity of the warehouse will increase. For Picnic, the throughput time of shipments is also important. A lower throughput time of shipments will mean that shipments can follow each other faster, increasing the system's performance. Workers are a big cost item, so needing fewer workers to realise the same output is good for performance. Hence, picker productivity is also an important performance metric.
The performance indicators are:

1. Timeliness of orders
2. Sojourn time of orders
3. Throughput time of shipments
4. productivity of pick process

These performance indicators will be the main output of the model as described in the next chapter.

4 Simulation model

The system will be modelled using a discrete simulation model. The model will include all elements mentioned in Section 1.2 and Section 3.1. In this chapter, the model description will be given. The model description will consist of a conceptual version of the model with an explanation of how the specific processes have been modelled in Section 4.1. The KPIs and how they have been implemented into the mmodel are discussed in Section 4.2. The model can be built using different Discrete Event Simulation tools. Section 4.3 aims to elaborate on the decision about the tool that will be used in this research. After the tool selection, Section 4.4 will describe the model implementation. The section will give a short overview of the model. A full technical model description will be given in Appendix A. Finally, Section 4.5 will also elaborate on all data points used within the model.

4.1. Conceptual model description

The problem, as described in Section 3.1, can be translated into a conceptual model. The conceptual model will cover all elements and processes that need to be in the eventual model and explain the model on a higher level. In Figure 4.1, a schematic overview of the system is shown. Using this schematic overview a conceptual model has been determined, this will be discussed later in this subsection. This schematic overview simplifies the system described in Section 3.1. In Figure 4.2, the conceptual model is shown.

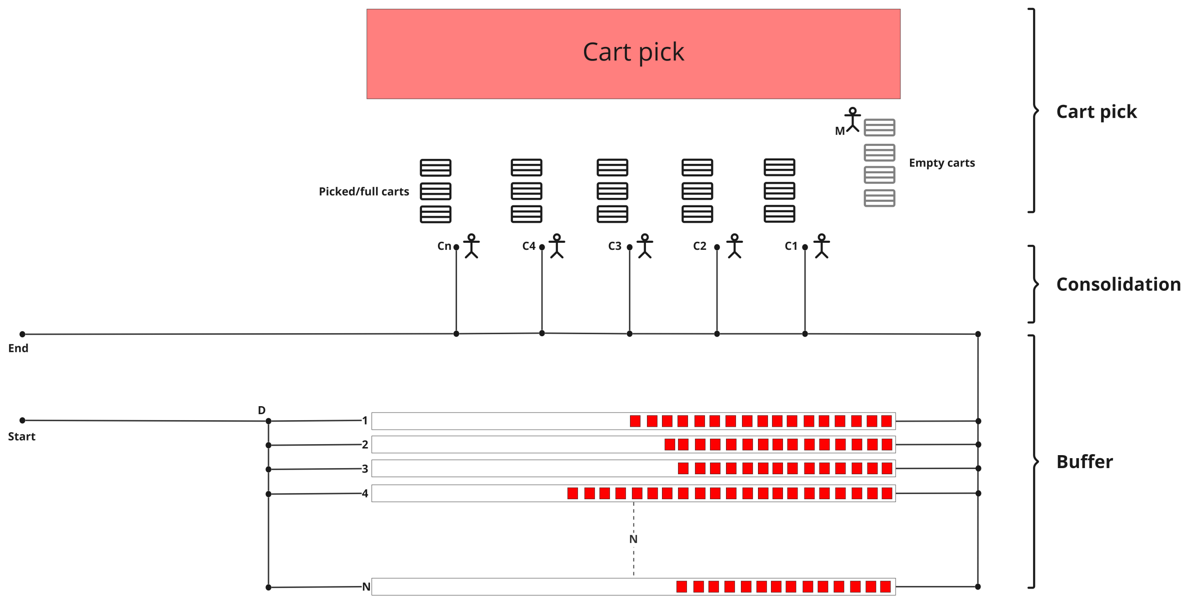

Figure 4.1: Schematic overview

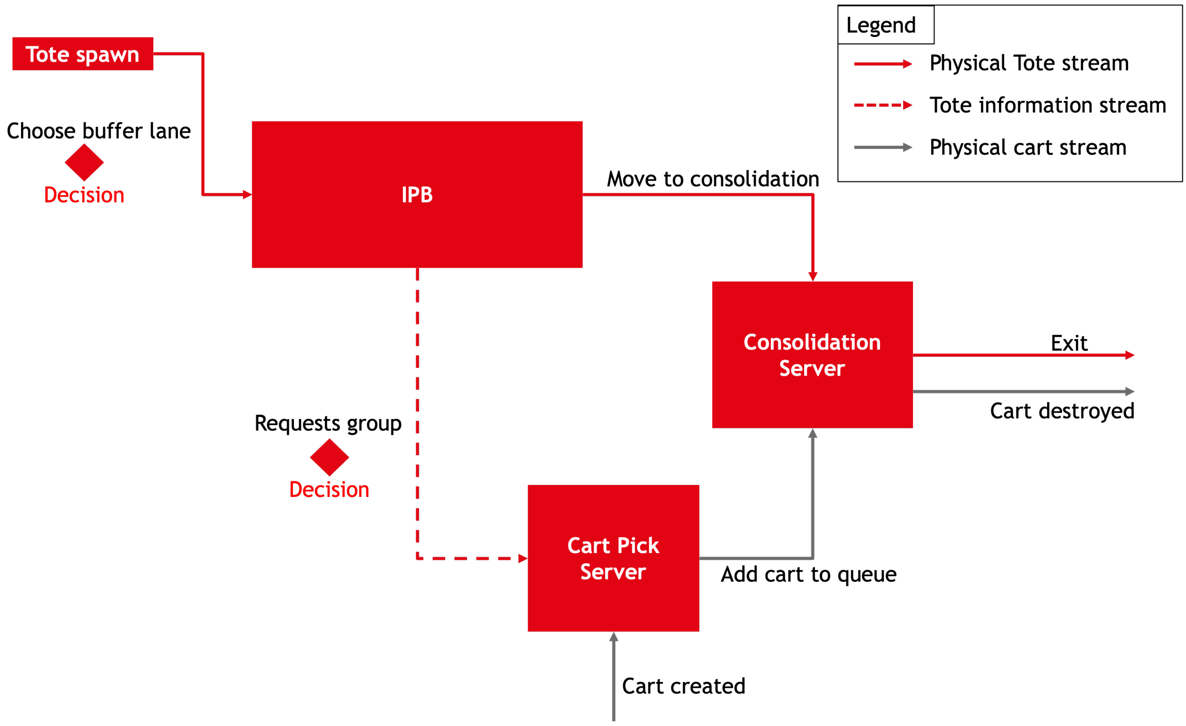

Figure 4.2: Conceptual model

As seen in Figure 4.2, all elements described in the problem description are included in the conceptual model. In this section, all elements and their processes will be explained. As can be seen, there are different streams defined. The solid red line is the physical stream of totes over a conveyor. The dotted red line is the information stream used by the cart pick server to group totes in the IPB together and start a pick round. Lastly, the grey line shows the physical stream of pick carts, which move on the ground, pushed by the pickers. In the model, the pick carts are simplified as an infinite amount because for this research the fact that there is a limited number of pick carts is not of importance. When a picker asks for a group and also gets assigned a group, a pick cart is created and after the picker is done the pick cart will leave the model.

The model starts with the tote spawn. In this part of the model, it is important to consider how totes would realistically appear in the system. Different data points can be used to model this part of the system, and it is important to use/create a valid dataset to run the model with. In Section 4.5, an elaboration about this data point will be described.

After the totes are generated, they go to the IPB after the spawn. After which, a decision will be made about where the tote needs to go. This is the lane selection as explained in subsection 3.1.1. Lane selection and the following group formation are important aspects of this research. The lane selection depends on different strategies that can be used and are defined in Appendix B. Totes have different properties, which will be based on real order data from the case study company Picnic. Totes will be assigned to buffer lanes based on their properties following different strategies. The properties that will be used are, for example, the deadline of an order, the frame number of an order and the truck ID of an order. After the tote is assigned to a buffer lane, it will move to that buffer lane.

While the tote is in the buffer lane, the cart pick server has different picker resources which will request a group. The group formation is explained in subsection 3.1.1. If a group can be formed, the picker will start the picking round with a duration based on the number of products in the totes in the group for which the picker is picking. This part of the model is also where variance disruptions will play a role. Disruptions like a picker going to the bathroom in the middle of the picking round, causing the round to take much longer than expected, will affect the model's outcome. After a picker completes the picking round, the picker puts the cart in the cart queue at the consolidation server.

At the consolidation server, there are multiple consolidation resources (workers), and if there is a picked cart in the queue, a worker will ask for the totes that are in the group of the picked cart. The totes will then move from the IPB to the consolidation station the worker is at, and the worker will start consolidating the order based on the number of products in the tote. After the consolidation, the totes leave the consolidation station and exit the system.

4.2. KPIs

From the literature in Section 2.2, KPIs used in literature for measuring the performance of a warehouse were given. With these KPIs and some other indicators mentioned in Section 3.1, the most important KPIs were discussed in a meeting with experts from Picnic. This meeting showed that the timeliness of orders is the most important. If an order is delayed, it will delay the whole shipment. Cancelling an order is only a last resort as this will have a big impact on customer retention which is very important.

For hybrid systems in general but also the system of Picnic, the sojourn time is also important. With a lower sojourn time of orders, more orders can pass through the system, and the system's capacity increases. This also means the order can be started later.

For Picnic, there is one specific KPI that should also be monitored. This is the throughput time of a shipment. For Picnic, if the throughput time of a shipment is low, it means that more shipments can be handled after each other. Another KPI that should be monitored for Picnic is the maximum number of items in the buffer. The buffer in the fulfilment centre in Dordrecht has a capacity of 22 totes. The maximum number of totes in the buffer should not exceed that maximum.

The KPIs and their formulas are listed in Table 4.1.

KPI name	Formula	
Timeliness of orders	$\sum_0^{Totes} t_{i,exit}\ where > t_{i,deadline}$	$\forall i \in Totes$
Sojourn time of orders	$\frac{\sum_0^{Totes} t_{i,exit} - t_{i,enter}}{N_{totes}}$	$\forall i \in Totes$
Throughput time of stack	$\frac{\sum_0^{Stacks} t_{j,0,enter} - t_{j,N,exit}}{N_{stacks}}$	$\forall j \in Stacks$
Productivity of pickers	$\frac{\sum_0^{Workers} picks_w / t_{w,worked} * 3600}{N_{workers}}$	$\forall w \in Workers$

Table 4.1: KPIs

Other things to remember when analysing the output could be to check whether orders with the same destination have exited the system around the same time. This would minimize the time that orders have to wait for a last order before the shipment can be shipped off.

4.3. Tool selection

According to the above conceptual model, different options exist for programming a simulation model. The first option is to use dedicated simulation software. Dedicated simulation software has the advantage that they have built-in solutions for most simulation studies. Moreover, most simulation software solutions have built-in extensive scenario analysis. Next to simulation software, it is also possible to program a simulation model using a programming environment. Python, for example, has different packages allowing an easy discrete event simulation environment setup. The advantage of this option is that it grants full freedom to the modeller to implement their logic into the model. Besides, using Python to program the model allows for easy data coupling and keeps the possibility of using mathematical models open.

For this research, Python will be used to program the simulation model. Using Python allows for easy data coupling. As the data from Picnic is in SQL databases, these tables can be read using SQL and saved to a CSV file which can be used in the simulation model. Another aspect of this decision is that Python is used within Picnic. Therefore, this will allow the model to be used by Picnic after the research is done. Additionally, Python is very flexible, allowing the modeller to be completely free to decide how the model and all aspects should work. As some aspects of the system are very specific, it could be that dedicated simulation software cannot replicate the exact behaviour or the exact strategy cannot be implemented. Next to that, Python also keeps the option for a possible connection to optimisation software open. A downside of Python is that every aspect of the model will have to be defined. In contrast, dedicated simulation software has many built-in features like servers, workers, and conveyors.

To not have to program a whole Discrete Event Simulation environment, different simulation packages are available in Python. PYDSOL, Salabim, and SimPy are some examples. Of these examples, SimPy is the most known and used. Within Picnic the Warehouse Simulation team uses SimPy to build a simulation tool for automated warehouses. Because of this, SimPy is chosen for this research. Being the most used package as well as being used within Picnic means support will be available online and through colleagues.

The main elements of SimPy that will be used are the simulation environment that allows for events and the future event list. Also, the resources will be used as that makes it easy to seize and release resources and model the different aspects of the system.

4.4. Model implementation

This section explains the most important workings of the model that has been created. The model incorporates all aspects of the conceptual model. Moreover, the model has a lot of things that improve its accuracy with how the system looks in the Dordrecht case. An extensive model description can be found in Appendix A.

4.4.1. parameters

This section will explain the input variables and how they are determined. The first two important input variables are the strategies for both the lane selection and the group formation. These can be changed as an input of the model and will impact the KPIs these input variables will help to answer the third research question. The second important set of input variables is the minimum and maximum group size. For each day a different minimum and maximum group size can and will be determined. Experiments with different values for these variables will show what values to use on different days. The number of pickers, consolidation stations and their processing times are also input variables which will impact the outcome of the system. They should also be varied to see their impact on the KPIs.

4.4.2. Orders

The arrivals are modelled using data from existing warehouses of Picnic. The justification can be found in Section 4.5. A dataset containing all orderlines of a specific day is obtained. Using a split of which orderlines will be in the manual pick area, which orders need to go to the manual pick area and how many products the orders still need are known. An orderline is the term for one article and is used for determining the efficiency of the picking process. This dataset contains all order information (truck id, (frame)stack id, deadline etc.), and is used in the model to determine when which tote enters the model and what the order information is. At every event the tote goes through, information is logged into a tote output file. This file will later be used in the analysis.

4.4.3. Lane selection

The next step in the model is the lane selection. The tote will need to enter the IPB and there are 20 different lanes to choose from. Different strategies will determine which lane the tote will be assigned to. The base strategy is the simplest strategy that can be used. This strategy will assign the tote to the lane with the least totes. The algorithm for this strategy is given in Code A.1. This base strategy will be the control to which other strategies will be compared.

Different strategies will have a different impact on the output of the model. If a strategy weighs heavily on putting together frame stacks, this would probably impact the average sojourn time of totes. A good strategy will balance all the different properties of the totes while not letting totes stay in the IPB for a very long time. Another aspect to consider in the strategies is the deadline of a tote. Totes that leave the IPB past their deadline will directly impact the timeliness of their shipment. The deadline of a tote is thus a very important aspect of the strategies. One solution to this could be to use one (or more lanes) for totes that are very short on their deadline. However, this impacts the fill rates of the other lanes and the workers' productivity.

4.4.4. IPB

After the tote is assigned to an IPB lane, it will enter that lane. At every event concerning the IPB, all IPB information is logged for later use in the analysis. Totes already assigned to a pick cart will stay in the IPB until a consolidation station is available and scans the pick cart belonging to that group.

One possible solution that come forward by analysing the system is to use an IPB lane as a priority lane. Using a priority lane is expected to decrease the number of totes that leave the model past their deadline. A priority lane works by implementing logic in the model that checks how much time is left until a tote's deadline when a tote enters the system. If this time is shorter than a certain threshold, the tote will get sent to a priority lane. When a picker requests a pick cart, and there are any totes in the priority lane, the picker will start a pick round for these totes.

4.4.5. Group formation

Group formation is another model aspect where different strategies will have different impacts. The group formation starts when a picker scans a pick cart. The group formation algorithm will then determine if and which totes will be assigned to the scanned pick cart, and the picker will start the picking round. The simplest group formation will choose the maximum number of totes from the IPB lane containing the most totes. Other strategies (as explained in Appendix B) will impact the KPIs, and the experiments will show which combination of lane selection and group formation will result in the system's optimal performance.

4.4.6. Cart pick and consolidation

After a group is selected, the worker that scanned the pick cart will start the pick round for that group. This pick round has a duration based on a constant time representing the walking around the circuit that always needs to happen. On top of that, for each orderline, additional time is added. This additional time is based on data from manual warehouses of Picnic which will closely resemble the duration of an orderline pick in this manual pick circuit. After the picker is done, it will put the cart in the queue at the consolidation area and be idle for a certain amount of time to model the picker ending the picking round and preparing a new cart. After this time, the picker will scan the new cart, and the group formation algorithm will assign a new group to the picker.

The consolidation workers will wait until a cart is in the queue with picked products for totes in the IPB. After scanning the full pick cart, the totes will move from the IPB lane to the consolidation station. At the consolidation station, the totes will appear in the consolidation position one by one. The worker will put the products from the slot in the cart containing the tote products in the consolidation position into the order tote. After the products of an order have been consolidated, the order tote will move out of the consolidation position and leave the consolidation station. This is also the end of the model. Before leaving the model, the tote logs its information into the output file. The pickers and consolidation resources also have an output file where they log their activities and specifications.

4.5. Input and configuration data

The model contains many data points which require research to get a realistic value for them. This section elaborates on how some of the data has been gathered. Because the system is new, some data points could not be taken from existing warehouses. To make good assumptions for these data points, experts within Picnic have been consulted and/or similar processes have been used. In Section A.2, a table with all data points and a justification for all data has been given.

The arrival of totes into the model is one of the most difficult things to model. Picnic does not yet have a hybrid fulfilment centre so no one-on-one data can be taken. The best approximation is determined after talking to people from Picnic's automated fulfilment centre (FCA). In the automated fulfilment centre, two processes are used to pick orders: zone pick (the same as in the hybrid fulfilment centre) and Goods To Person (GTP). A representative distribution of tote arrivals can be obtained by looking at the distribution of timestamps that totes have their last pick done in zone pick in FCA, see Figure A.1. To approximate this distribution, a lognormal distribution is drawn that closely resembles (see Figure A.2) the distribution of the timestamps of FCA. The order information is something that can be easily taken from existing order data. Thus, it will be very realistic as to what it will look like in the hybrid fulfilment centre. Also, it is known how the split between what products will be in cart pick and what products will be in zone pick. Using this information, the number of products that need to be picked in the manual pick area can be determined from the order info. This split is explained further in Appendix A

Two other important data points are the pick times and the consolidation times. For the pick times, an estimation is made for the standard time it takes for a picker to walk along the route. Using data from manual warehouses of Picnic, the extra time per orderline is determined. Appendix A details how this value is determined. The value for the consolidation time is based on data from the highly automated fulfilment centre. A triangular distribution is used for these values as there is a clear minimum, maximum, and mean process time these processes take.

The base input that will be used for the experiments is given in Table 4.2.

Table 4.2: Base input

Parameter	Value
buffer lanes [#]	19
consolidation time [minutes]	
- mean	12
- min	8
- max	6
entry strategy	base
group strategy	base
max group size [#]	15
min group size [#]	5
number pickers [#]	18
number stations [#]	6
pick time: base round [seconds]	250
pick time: orderline [seconds]	
- mean	14.4
- min	12
- max	18
priority lanes [#]	1
consolidation outage time [minutes]	0

5 Validation and verification

Validation and verification tests will determine if the model outputs expected values and if the model works as expected. This chapter will present the validation and verification of the final model. This will be done through various tests. The most important tests will be explained in this chapter. The full analysis and other tests are in Appendix C.

With model validation, the goal is to validate if the model works as expected. That means if something is expected to happen in the natural system, it should also show this in the model. Validation can be tested in many different ways and on many different scales. The outcome of the whole model can be tested or a single entity in the model is validated as expected. Some tests are designed for validation; this section will present the result. As the fulfilment centre in Dordrecht is not yet operational there is no data for the system's output. Thus, the validation will have to be done by validation through face validity using the knowledge of experts within Picnic. This is the way that the validation tests have been done.

5.1. Total performance

As stated above, the system is not yet operational so validation will be done by comparing it to the design values obtained from the experts within Picnic that designed the system. Also, the system's behaviour expected by these experts will be used for validation. To test if the total performance of the model corresponds to the design in Dordrecht, it will be tested whether the model fails if we use the full dataset (close to the maximum that the system can handle as designed by Picnic) and take out a part of the system. In this test case, the full dataset will be used and take away one consolidation station to see if the model also shows that this will result in an out-of-balance situation. The results are shown in Figure 5.1 and Figure 5.2. The first five experiments have six consolidation stations, so all consolidation stations, and 17 through 21 pickers. The next five experiments have five consolidation stations, so one station missing, and the same number of workers. Figure 5.1 shows the number of late totes if both experiments and Figure 5.2 shows the average sojourn time of both experiments.

As can be seen, the system cannot keep up with the number of totes entering the system when a critical aspect is missing. After talking to experts at Picnic, they also confirmed that the system is designed to handle the peak day volume with all aspects of the system working. For Picnic, this also means that all six consolidation stations must work on a peak day. This validation experiment shows that the model's overall performance works as expected. On a peak day, when there are enough workers, the system is healthy, but when one-sixth of the consolidation capacity is missing, the system cannot keep up anymore. Interestingly, the number of pickers in a situation with 5 consolidation stations seems to impact the late totes.

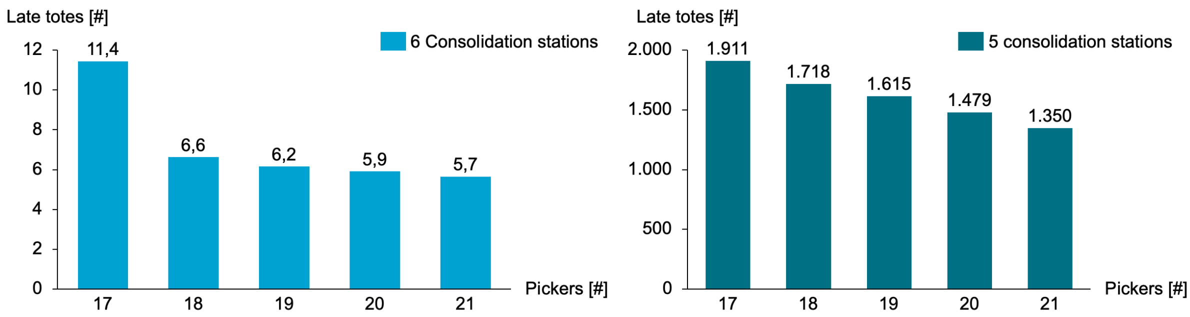

Figure 5.1: Number of late totes: 5 and 6 consolidation stations

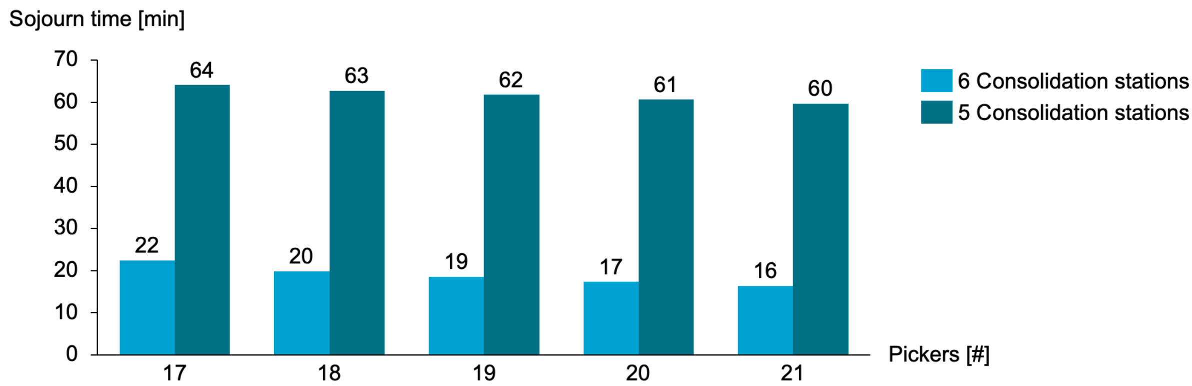

Figure 5.2: Sojourn time: 5 and 6 consolidation stations

5.2. Number of pickers

When the number of pickers in the model increases, the number of late totes is expected to decrease gradually, and the sojourn time will decrease significantly. While the totes from the normal IPB lanes will be picked faster, the late totes still receive priority. Hence, the number of late totes probably will decrease but not as much as the sojourn time. However, picker productivity will also decrease, which is inefficient. For picker productivity, pickers are expected to do pick rounds for smaller groups of totes. This means that their productivity will go down. For this validation test, the number of pickers is increased from 18 to 27. These results are shown in Figure 5.3 and Figure 5.4.

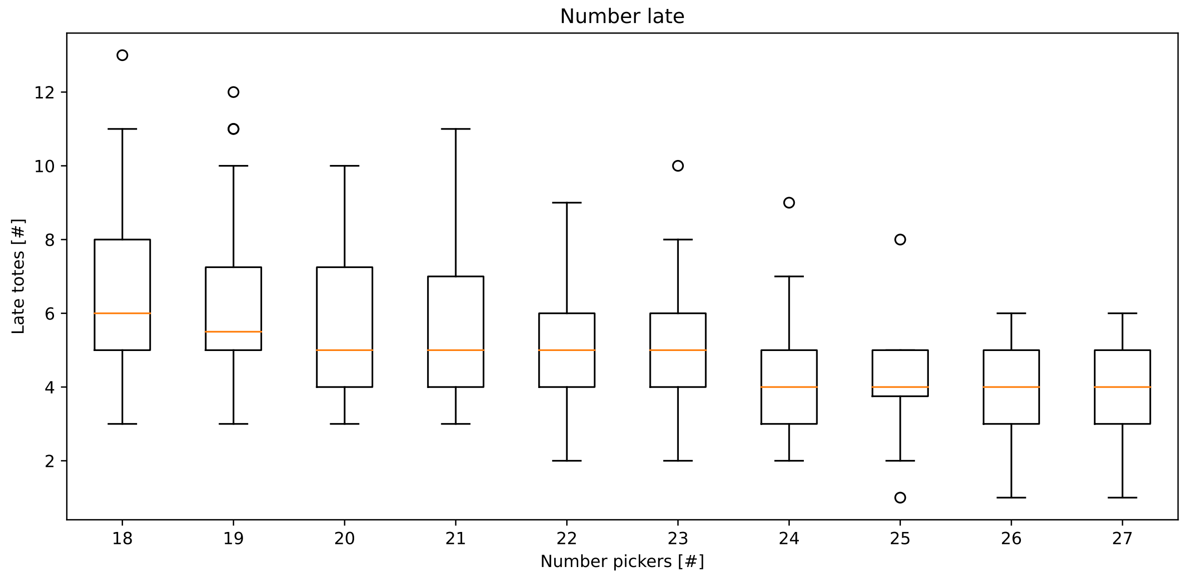

Figure 5.3: Number of pickers validation - Number late

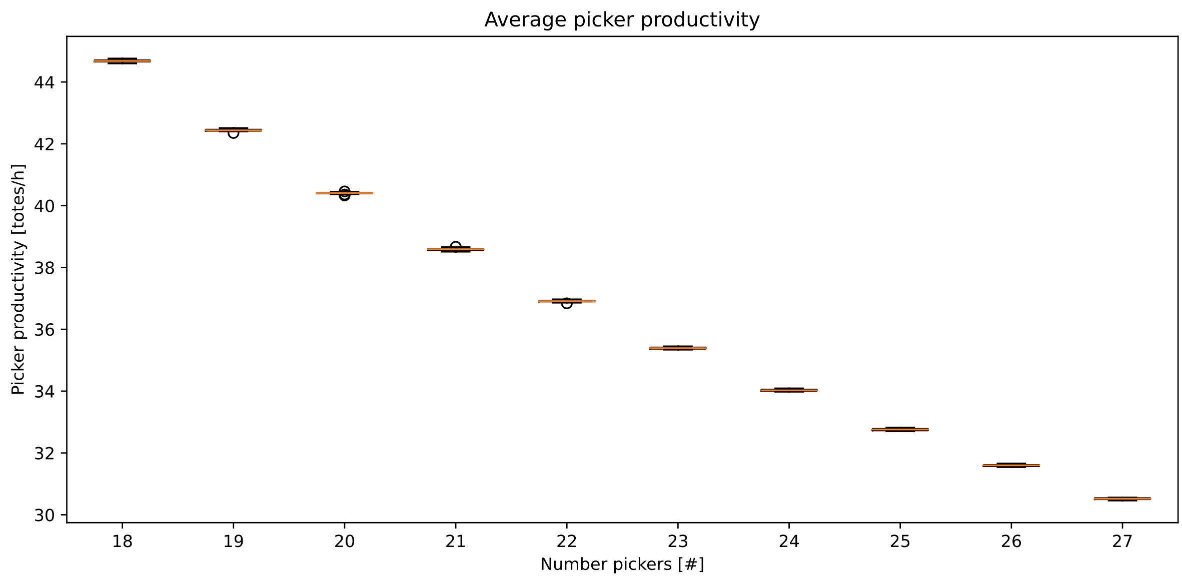

Figure 5.4: Number of pickers validation - Sojourn time

As expected, the number of late totes does decrease but not significantly. The average sojourn time and average picker productivity decrease a lot more. It can also be seen that the values for average sojourn time decrease with smaller amounts as the number of pickers increases. This is also expected as there comes a time when the sojourn time cannot decrease any further. In this system, after the totes enter the IPB, a picker will usually be ready to start a picking round immediately. This picking round will likely also be for a smaller group so that it will be done faster. Hence, this explains why the sojourn time can get so low.

6 Experiments

To find out what the best strategies should be used and what the best configuration for the manual pick process is, experiments will be defined. In this chapter, the experimental design will be explained. All experiments conducted are described in Appendix D. In this chapter, in Section 6.1 the experimental setup is explained. In Section 6.2, the experimental designs for experiments with interesting results are explained.

6.1. Experimental setup

From the results of Section D.1, 32 replications give reliable results with narrow confidence intervals. This is the number of replications that will be used for experiments. One model run consists of one day. As order information, a dataset from Picnic is used, which contains correct order data for a peak day of the fulfilment centre in Dordrecht. Also, a small order file is available to simulate a less busy day.

Experiments are run across multiple replications to eliminate random occurrences in a model. Replications ensure that a model run is not an outlier across other model runs and that accurate conclusions can be drawn from the model results. In this model, 32 replications are enough. In Appendix D, the analysis to find the number of replications is given.

The replications are run with a different random seed for the experiments. To keep the comparison between experiments as reliable as possible, the random seed is the same for each replication number in each experiment. Table 6.1 shows how the model's output will be summarized for each experiment. The average, minimum, maximum, and half-width are given for each statistic. This half-width is calculated with a 95% confidence interval.

Table 6.1: Example output

Deadline min & IPB time	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	-	-	-	-
	Maximum	-	-	-	-
IPB time (min)	Average	-	-	-	-
	Maximum	-	-	-	-
Number late (#)	Total	-	-	-	-
stack throughput time (min)	Average	-	-	-	-
	Maximum	-	-	-	-
picker productivity (#/h)	Average	-	-	-	-
	Minimum	-	-	-	-
	Maximum	-	-	-	-
buffer items (#)	Maximum	-	-	-	-

6.2. Experimental design

To make valid statements about the effectiveness of different configurations of the system, a good experimental design has to be made. In the experimental design, the values for the input will be varied in such a way that the output will be able to show what impact different configurations of the system have on the output. Also, what combination of inputs will give the most desired output and what the trade-offs are.

The base input for all experiments is explained in Section 4.5. After the strategies have been analysed, good-performing strategy combinations will be used in the rest of the experiments. After the base experiments have been done for all categories, combinations of configurations will be analysed to find other combinations that positively impact the system's performance.

6.2.1. Buffer strategies

The buffer strategies experiments have been done regarding the priority lane and the strategy combinations. The next two sections will explain the experimental setup for these experiments.

Priority lane

To find out if a priority lane positively impacts the system's performance, the first experiment that will be conducted compares the performance of a system without a priority lane with the performance of a system with a priority lane. As the strategy is expected to influence the outcome, this experiment is performed with all strategy combinations. If a priority lane has a big positive influence on the system's performance, a priority lane will also be used in other experiments.

strategy combinations

For the strategies, it is essential to vary the lane selection and group formation strategies and find if any combination will give superior results and if any specific strategy outperforms others. The strategies are extensively explained in Appendix B.

For the strategies, the experimental design will combine all lane selection strategies with all group formation strategies. As the strategies are essential for the research, finding the best combination of strategies is very important. Estimating possible synergies between different strategies may be difficult, so analysing all combinations eliminates missing possible good strategy combinations. The design is displayed in Table 6.2. The other experiments will also use the combination that gives the best and most consistent results. After all the separate experiments have been done, different combinations of experiments with other strategies will be performed.

Table 6.2: Experimental design: Strategies

Ex#	Lane selection strategy	Group formation strategy
S1	Base	Base
S2	Min groups	Base
S3	Min one by one	Base
S4	Max groups	Base
S5	Shipment Min group	Base
S6	Shipment Max group	Base
S7	Stack Min group	Base
S8	Stack Max group	Base
S9	Deadline Min group	Base
S10	Deadline Max group	Base
S11	Base	IPB Time
S12	Min groups	IPB Time
S13	Min one by one	IPB Time
S14	Max groups	IPB Time
S15	Shipment Min group	IPB Time
S16	Shipment Max group	IPB Time
S17	Stack Min group	IPB Time
S18	Stack Max group	IPB Time
S19	Deadline Min group	IPB Time
S20	Deadline Max group	IPB Time
S21	Base	Total deadline
S22	Min groups	Total deadline
S23	Min one by one	Total deadline
S24	Max groups	Total deadline
S25	Shipment Min group	Total deadline
S26	Shipment Max group	Total deadline
S27	Stack Min group	Total deadline
S28	Stack Max group	Total deadline
S29	Deadline Min group	Total deadline
S30	Deadline Max group	Total deadline
S31	Base	Single deadline
S32	Min groups	Single deadline
S33	Min one by one	Single deadline
S34	Max groups	Single deadline
S35	Shipment Min group	Single deadline
S36	Shipment Max group	Single deadline
S37	Stack Min group	Single deadline
S38	Stack Max group	Single deadline
S39	Deadline Min group	Single deadline
v40	Deadline Max group	Single deadline

6.2.2. Manual picking

For manual picking, different values can be experimented with. The number of pickers will most likely significantly influence the model's performance, which is also a value that should be optimised through the experiments. The group size will also influence the KPIs in different ways, and the impact should be analysed. Besides, the group size will most likely impact the performance differently for different strategies, so this combination should also be tested. Lastly, an interesting experiment will see if these findings change when the small dataset is used so the model simulates a less busy day. These experiments have all been performed, and their experimental designs can be find in Section D.3. After analysing the results of the experiments, it was expected that the group size and the strategy would influence each other, so an experiment was defined for testing those combinations. This is expected because the strategies also take the group size into account. The experimental design is given in Table 6.3.

6.2.3. Disruptions

From earlier experiments, it could be seen that for the Dordrecht warehouse, on a busy day, the system would be unable to keep up with the incoming totes if one consolidation station was missing for the whole day. Even though this makes sense, as the system is designed to be able to handle a peak day with all of its capacity. However, analysing how long the system can operate with one broken consolidation station can give interesting insights. Results from these experiments can point out how long the maintenance operator would have to fix the problem before the system would end up in an unrecoverable state. This experimental design is shown in Table 6.4.

Table 6.3: Experimental design: group size and strategy

Ex#	Lane selection	Min group size	Max group size
G1	Base	3	12
G2	Min groups	3	12
G3	Min one by one	3	12
G4	Max groups	3	12
G5	Shipment Min group	3	12
G6	Shipment Max group	3	12
G7	Stack Min group	3	12
G8	Stack Max group	3	12
G9	Deadline Min group	3	12
G10	Deadline Max group	3	12
G11	Base	3	19
G12	Min groups	3	19
G13	Min one by one	3	19
G14	Max groups	3	19
G15	Shipment Min group	3	19
G16	Shipment Max group	3	19
G17	Stack Min group	3	19
G18	Stack Max group	3	19
G19	Deadline Min group	3	19
G20	Deadline Max group	3	19
G21	Base	10	12
G22	Min groups	10	12
G23	Min one by one	10	12
G24	Max groups	10	12
G25	Shipment Min group	10	12
G26	Shipment Max group	10	12
G27	Stack Min group	10	12
G28	Stack Max group	10	12
G29	Deadline Min group	10	12
G30	Deadline Max group	10	12
G31	Base	10	19
G32	Min groups	10	19
G33	Min one by one	10	19
G34	Max groups	10	19
G35	Shipment Min group	10	19
G36	Shipment Max group	10	19
G37	Stack Min group	10	19
G38	Stack Max group	10	19
G39	Deadline Min group	10	19
G40	Deadline Max group	10	19

Table 6.4: Experimental design: Consolidation outage

Ex#	Consolidation outage	Pickers
1	0	18
2	30	18
3	60	18
4	90	18
5	120	18
6	150	18
7	180	18
8	210	18
9	0	22
10	30	22
11	60	22
12	90	22
13	120	22
14	150	22
15	180	22
16	210	22

7 Results

In this chapter, the most interesting and important results will be discussed. The full analysis of all experiments can be found in Appendix E. subsection 7.1.1 gives the results of the experiments regarding the priority lane. subsection 7.1.2 explains the results of the experiments for the strategies for the lane selection and group formation. Section 7.2 highlights the results of the experiments for the configuration of the manual pick section. Lastly, Section 7.3 shows the results of the experiments for the disruptions in the manual process.

7.1. Buffer strategies

The buffer strategies experiments have been done regarding the priority lane and the strategy combinations. The next two sections will explain the results of these experiments.

7.1.1. Priority lane

Using a priority lane will significantly improve the performance of the warehouse. As seen from Figure 7.1, the number of late totes decreased by 83%, which is the most important KPI. It was expected that the sojourn time would increase as totes with no priority would have to wait for totes with priority. Also, the picker productivity was expected to decrease as pick rounds for a priority tote do not have a minimum group size. Thus, a picker could start a pick round for a single tote. The results show that the average sojourn time increases by 5%. This is quite an increase, but it does not compare to the 83% decrease in late totes which is a more important metric and shows a bigger improvement. For the picker productivity, the decrease is very small (0,2%). This can be explained by the small number of totes that actually need priority service compared to the large number of normal pick rounds. The decrease in picker productivity should thus not be a limiting factor in deciding a priority lane.

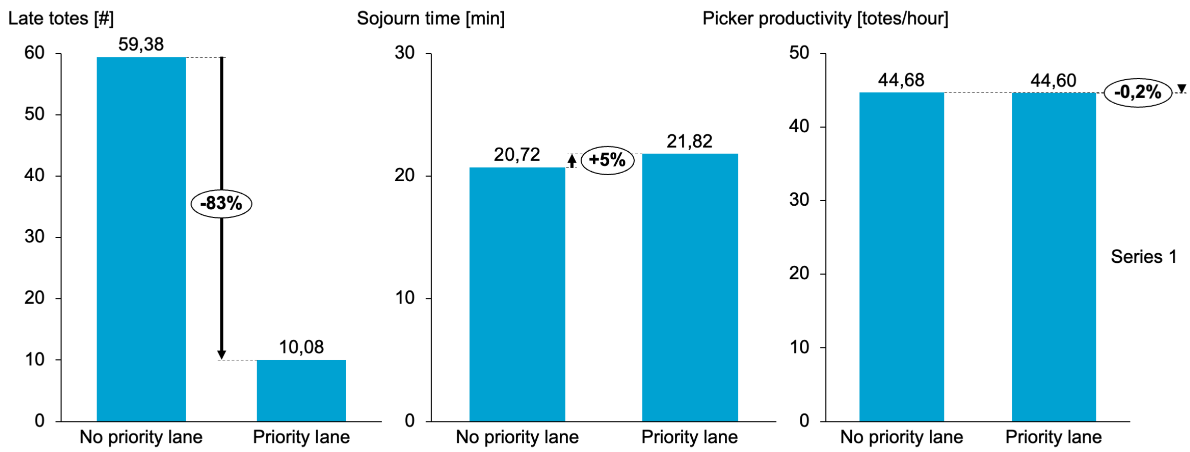

Figure 7.1: Results priority lane

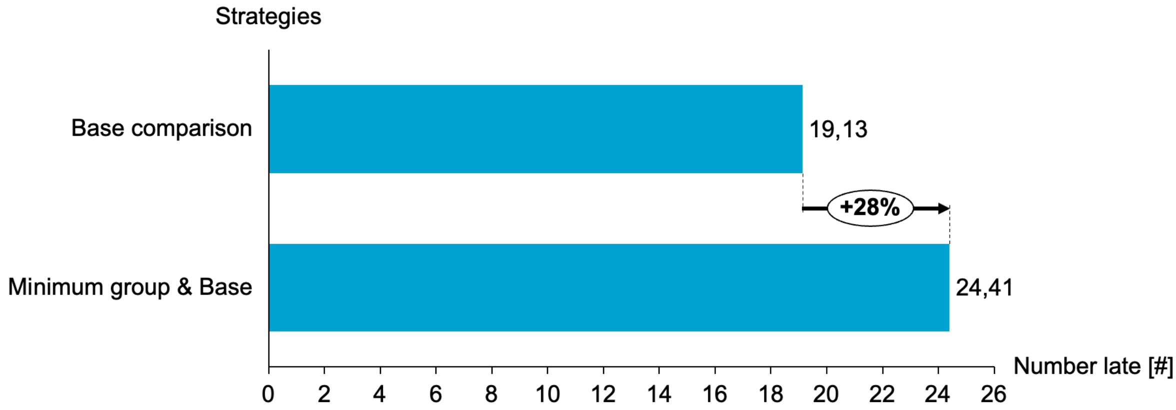

Figure 7.2: Results base and minimum group size

From the priority lane experiment results, it was decided to use a priority lane in the system for the remaining experiments.

7.1.2. Strategy combinations

After the full factorial experiment for the combinations of strategies, the best strategy combinations became clear. Combining the minimum group size strategy for lane selection with the base strategy for group formation gave worse results than using the base strategy for both processes, as can be seen from Figure 7.2. Using a different strategy was always expected to improve performance, but it can be explained after looking at this strategy combination. The minimum groups strategy fills the IPB lanes to the minimum group size first and to the maximum group size after. The fact that the base strategy for group formation chooses the lane with the most totes likely results in the last lanes of the IPB containing totes for a very long time. This likely results in a higher number of late totes.

What could immediately be seen from the results was that two strategy combinations outperformed other combinations. These two combinations are:

deadline min group & IPB Time
deadline max group & IPB Time

Figure 7.3 shows these results together with the third-best strategy, which also performs better than the base comparison. The base comparison uses the base strategy for lane selection and group formation, and as said in subsection 7.1.1, these experiments also included a priority lane.

The deadline seems to be the most important property to prioritise for the lane selection strategy. This can also be seen from Figure 7.3, as the deadline and the shipment are very closely correlated. One important finding is that the combination of lane selection based on the deadline and group formation based on the deadline doesn't give better results. In most cases, this combination performs worse than prioritising the deadline in the lane selection. This is interesting as it could be expected that if the deadline is an important property to prioritise, it will show for both lane selection and group formation. From this, it seems that it is already enough when considering the lane selection deadline and prioritising it for the group formation as well performs worse. Other performance indicators will start to be neglected in this case causing this combination to be worse than using the IPB time for the group formation.

7.2. Manual picking

From the group size experiments, there appeared to be a trade-off for the minimum group size. Using a higher minimum group size results in more late totes but the picker productivity increases. This can be seen from Figure 7.4. As can be seen from the figure, the number of late totes starts with a small increase but increases faster as the minimum group size becomes higher. Thus, careful consideration must be made to determine the desired minimum group size, ensuring a certain picker productivity.

The group size is also tested and compared to different strategies. This experiment also gave some interesting results. A small minimum group size is not preferred in combination with a small maximum

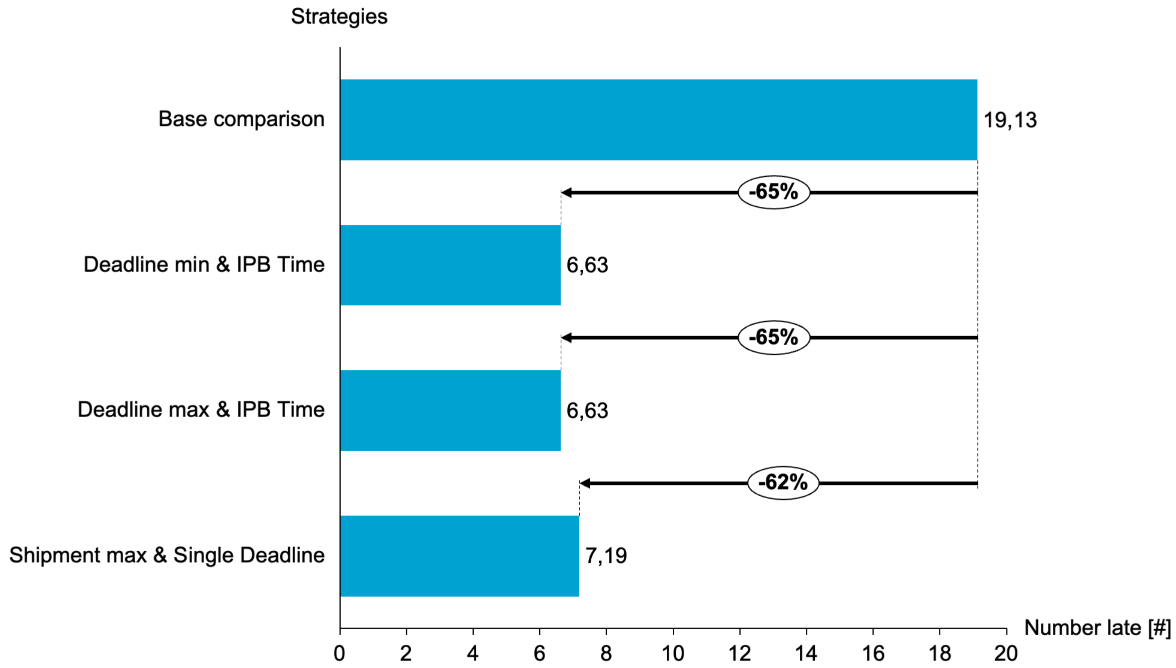

Figure 7.3: Results stratagies

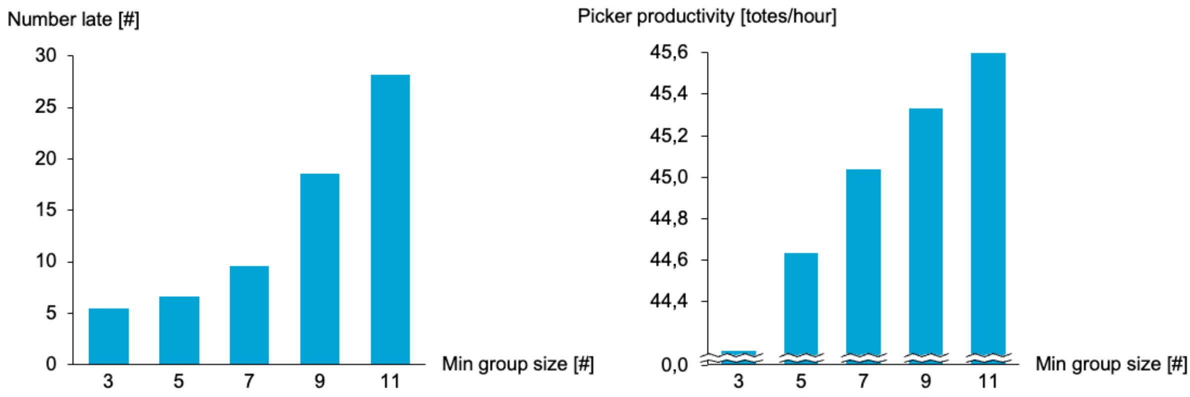

Figure 7.4: Results minimum group size

group size as it results in more late totes across all strategies. When looking at the number of late totes, combining a small minimum group size with a large maximum group size gives the best results. One thing to note is that the picker productivity will be lower with a smaller minimum group size. Hence, this is a trade-off that should be considered when deciding the values for the minimum group size. From these results, the maximum group size should be as high as possible. For Picnic, for example, this would mean that creating a big pick cart for big groups would positively impact the performance.

For a less busy day, the strategies prioritising the minimum group size were expected to outperform other strategies. Figure E.25 to Figure E.28 from Appendix E also showed this. Still, a low minimum and a large maximum resulted in the least late totes. So for a less busy day, it will again be best to set the maximum group size as high as possible, and the minimum group size should be decided based on preferred picker productivity. for a less busy day, it is recommended to use a lower minimum group size than on a busy day.

The number of pickers should be balanced, as there is a tipping point where the late totes will shoot up. After that tipping point, the number of late totes no longer decreases when the number of pickers increases. Still, the other KPIs perform better except for the picker productivity, which decreases as the number of pickers increases. Therefore, a trade-off between enough workers and the high preferred productivity should be made. Figure 7.5 shows that the number of pickers should be carefully selected as the number of late totes doesn't decrease significantly after 18 pickers. Still, productivity goes down linearly, which is to be expected. Having more workers than minimally necessary will thus be necessary but there is a trade-off in productivity when deciding how many more workers than necessary will be planned.

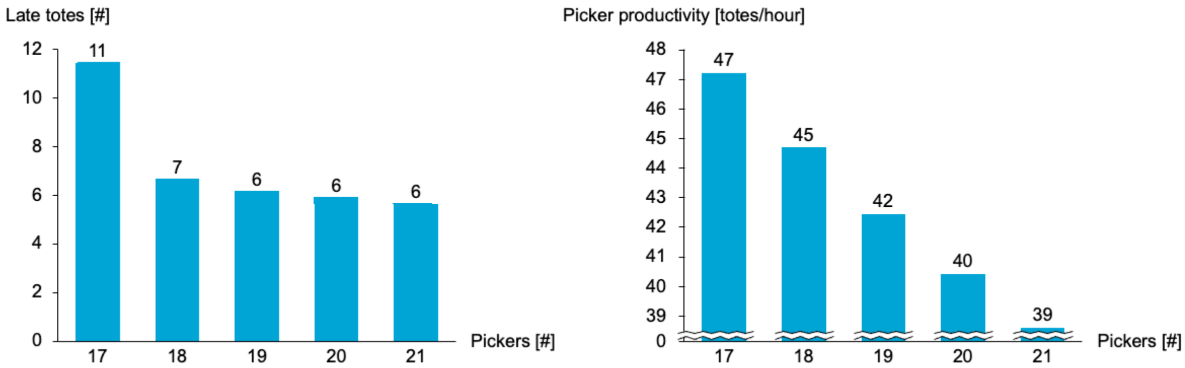

Figure 7.5: Results number pickers

7.3. Disruptions

From the experiments for the cart pick section of the system, an interesting result came from the experiments where the number of pickers and consolidation stations was varied. When using the full dataset, missing one consolidation station resulted in the system being unable to keep up with the totes entering the system, and the buffer would overflow. The facility is designed to handle a peak day of orders with all its elements, so this makes sense. One new experiment emerged from this finding. This experiment breaks a consolidation station in each run for a specified time. The results then show how long a consolidation station could be missed without the system's inability to catch up. The buffer lanes in the case study have a capacity of 22 totes, so when the maximum buffer items go above 22, the buffer will overflow, and the processes preceding the studied system will be affected. These experiments show that if there are enough pickers, the system can catch up if the consolidation station is repaired within 90 to 120 minutes. This means that if a consolidation station breaks on a peak day, it would have to be repaired within 90 to 120 minutes for the system to recover again. Otherwise, the buffers would overflow, causing problems in the zone pick area of the fulfilment centre, and the number of late totes would become too high. This can be seen in Figure 7.6.

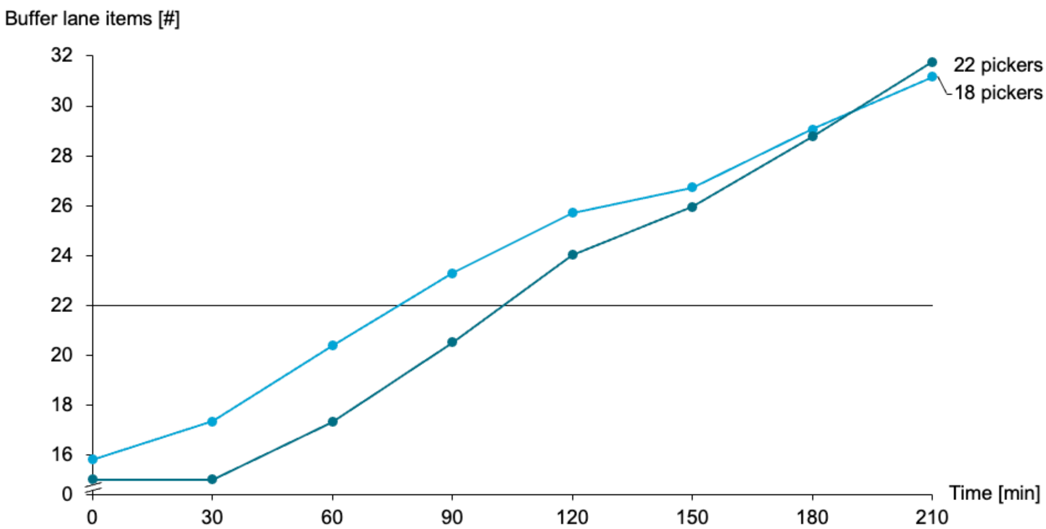

Figure 7.6: Results consolidation outage

8
Discussion

In this chapter, the discussion will restate the main findings and interpretations from the results. Additionally, the main limitations of the research will be discussed. Section 8.1, summarises the most important results. Section 8.2 highlights the practical implications for Picnic, while Section 8.3 highlights the general implications for warehouses. Finally, Section 8.4 discusses the limitations of this research and suggests ideas for future research.

8.1. Results

This research aims to find the best allocation strategy and group formation combination for the buffer and the best configuration of the manual picking process that follows after the buffer. From the results, it becomes clear that the combination of prioritising the deadline for the lane selection and prioritising the IPB time for the group formation gives the best results. This is surprising as it could be expected that if the deadline would be important in the lane selection, it would also be important for the group formation. It seems that if the lane selection focuses mainly on the deadline, it is unnecessary for the group selection strategy to consider the deadline. The group formation could best use another strategy to ensure that other totes would not wait too long in the IPB, thus the IPB time.

One thing that came forward from the experiments for the priority lane is that, as expected, the number of late totes decreased, but the sojourn time increased, and the productivity decreased. The results show that the decrease in late totes was significantly higher than the decrease in productivity and increase in sojourn time. This resulted in the conclusion that one IPB lane should be used as a priority lane.

Another key insight from the results is in line with the expectation. For the group size, the results show that it is better to use a larger minimum group size on a busy day. In contrast, the minimum group size should be lower on a less busy day. Hence, this parameter should be changed based on the expected number of orders that will come in on a specific day. Combining the group size with a lane selection strategy focussing on filling IPB lanes to the minimum group size will give the best and most robust results for busy and less busy days. On busy days lane selection strategies focusing on filling to the max group size will also give good results. It might be a good idea to balance the group sizes on busy days to allow for good picker productivity.

8.2. Practical implications for Picnic

This section will translate the main findings of the results into practical implications for Picnic. For Picnic, the IPB in this system is very new. In their highly automated fulfilment centre, there is a buffer but it works very differently. Hence, the results from this simulation model can give Picnic valuable insights into how to organise the control strategies of the buffer.

Beginning with the lane selection, from all experiments that are conducted with the strategies, prioritising the deadline showed the best results. This makes sense as the deadline is the most important KPI from interviews with Picnic employees. After prioritising the deadline, buffer lanes should be filled with minimal groups. When the system runs at higher volumes filling the buffer lanes to the max group size after prioritising the deadline improves picker productivity. For group formation, selecting the group of totes with the highest total time in the IPB will be best. Using the highest total IPB time resulted in the best values for the KPIs.

Next to the implications for the buffer, some findings are also important for Picnic regarding the configuration of the manual pick process. It is important to operate with slightly more pickers than needed to fulfil the orders everyday. For the consolidation stations, it is important to note that on a busy day (close to maximum capacity), a consolidation station cannot be missed for very long. Thus, a maintenance operator should be on standby to repair the consolidation station within 60 to 90 minutes if one breaks down on a busy day.

8.3. General practical implications

From Chapter 1 and Chapter 2, general research into buffer allocation of orders appears missing in the literature. From this research, some findings can be generalised to fill these knowledge gaps.

First of all, it seems that using efficient allocation strategies can significantly improve the performance of the studied system. This means that one takeaway is that thinking of good strategies for allocating orders into a buffer and strategies on how to take orders out of the buffer is of great importance. This can impact both the timeliness of orders as well as the total throughput of the system and how fast orders go through the system. From the literature in Section 2.6, research into the allocation of orders into a buffer was missing.

Also, it appeared that primarily focusing on the deadline of an order would be best for lane selection. So when allocating orders to a buffer, it will be best to bunch them up by their deadline. For taking orders out of a buffer, it appeared to be best to focus on the total time in the buffer of the orders.

Not every system will have groups of orders handled in the last process. Still, if there are groups of orders, it is recommended to distinguish between the minimum group size for busy days and a different minimum group size for less busy days. This creates the perfect balance between wait time in the buffer and the productivity of the worker/process after the buffer.

If timeliness is an important KPI, it will always be a good idea to sacrifice buffer capacity to have one lane as a priority lane. Orders close to their deadline can use this priority lane and be completed without waiting.

8.4. Limitations and future research proposals

The model has some limitations, and interesting limitations can be further investigated in future research. Some aspects of the system have been reduced or simplified to fit the scope into the research timeline. This section will point out suggestions for future research and details on improving the model.

As the model is a simulation model, all aspects are defined by the modeller. Hence, the problem is not 'optimised'. By analysing the system, the tote properties and possible combinations, many possibilities were covered to draw a valid conclusion. However, the problem can not be called optimised.

For future research, it is advised to see if the problem could be formulated into an optimisation problem. It might be that some properties have not been taken into account, but true optimisation might give good results. One downside to a true optimisation model might be that it is hard to formulate the outcome of an optimisation model into strategies. Letting an optimisation model find the best lane selection and group formation could result in combinations that cannot be formulated into clear strategies to use with different days of orders.

For the input data, the totes are spread on the shipment level. As one of the KPIs tries to minimise the stack throughput time, it might be best for Picnic to try and keep the spread of the stacks in the system as low as possible. This spread will likely be too large in the model as the totes are spread on shipment instead of stack level.

9
Conclusion

In this chapter, the objective is to consolidate the findings and provide comprehensive responses to the main research question and the sub-questions. The chapter will begin by addressing the sub-questions and subsequently presenting a conclusive answer to the main research question.

The primary objective of this research is to bridge a gap in the existing literature by exploring the allocation of totes to a buffer in a hybrid warehouse that combines two processes. While previous studies focus primarily on optimizing the buffer size, this research aims to investigate the optimal allocation of orders within the buffer. The orders and the system possess various properties and parameters that can significantly impact the system's overall performance. As such, this study sought to examine and analyze these parameters to enhance the understanding of the system's total performance.

This research aims to find buffer strategies for allocating orders to a buffer lane and selecting groups from the buffer to start a manual pick round. Different combinations of buffer strategies impact the system differently, and strategy combinations should work under different circumstances. Another aspect of this research is configuring the manual picking process that follows the buffer. This process involves humans picking the products for an order and consolidating the order, after which the order tote leaves the system.

In literature, different methods are used for researching buffer strategies: optimisation models and simulation studies. An optimisation model can be defined in different forms. The mathematical model might be infeasible because of the complex nature of this specific problem emerging from the different properties and possible combinations in the buffer. After consideration, it has been decided to use a simulation model for this research.

The main research question for this research is:

> *'What set of strategies for the input buffer and the manual picking process is most effective for improving the performance of a hybrid warehouse system?'*

The following sub-questions have been established to aid in answering the main research question:

1. What are the performance metrics of hybrid warehouse systems?
2. What are the possible strategies for the input buffer?
3. What are the possible configurations of the manual process?
4. What is the influence of strategies for the buffer on the performance metrics?
5. What is the influence of different configurations for the manual process on the performance metrics?
6. What are the effects of disturbances on the performance metrics?

Most important performance metrics

To identify the most important performance metrics of the system, some metrics are found in the literature. Other metrics are found in the analysis Picnic uses to measure the performance of their systems. Experts within Picnic are conducted to find the final performance metrics, and the performance metrics are established. These performance metrics are the Number of late totes, Average sojourn time, Average stack throughput time and Picker productivity. The number of late totes is the most important for Picnic and should drive the decisions on what configurations to use. Late orders can cause customer impact, which costs Picnic money and customer retention. The other KPIs should also be optimised, as they improve the system's performance and can thus increase the financial impact for Picnic.

Possible strategies for the input buffer

The second question is about the strategies. These have been determined by analysing the system. Also, discussions with Picnic employees give properties that could be used in the strategies. Strategies have been determined for lane selection and group formation. The strategies are an important aspect of the system as one of the goals of this research is to find out if strategies can improve the performance of a warehouse. Strategies for the lane selection focused on the number of totes in a lane. for the tote properties they focused on the shipment and stack ID of a tote as well as the deadline of a tote.

From the research it could be concluded that for different warehouses different properties and thus different strategies will be formed. The research also showed that these strategies could improve the performance of the warehouse. This shows that it is advised to think of good strategies with the properties at hand.

Possible configurations of the manual process

The third questions is about the configuration of the manual process. What are they, and how do they impact the performance metrics? While creating the model, these parameters are identified. Also, discussions with Picnic employees give parameters that could be implemented into the model to affect different aspects. The most important parameters that have also been experimented with are the minimum and maximum group size, the number of workers in the manual process, the pick times and their variability, and the input data. These are the most important parameters as they are in control of the operator of the system. The input data, the number of orders that enter the system and their property, is also determined in this model. The number of orders and their properties is always known in advance for Picnic. Hence, Picnic has control over how they start new totes in the system, and it is known in advance whether it will be a busy or less busy day.

Effective strategies for improving the performance metrics

For answering the fourth sub-question, the results of the experiments are important. It is concluded that taking the deadline of the totes into account for the lane selection results in the best outcome for the KPIs. When this is combined with the group strategy based on IPB time, the system's performance is at its best. Combining a lane selection strategy based on the deadline of totes with a grouping strategy based on the deadline results in worse outcomes, which is interesting. It appears that when the deadline is taken into account for the lane selection, there is no added value in taking it into account with the lane selection. Rather selecting groups based on their time in the buffer will give better results for almost all KPIs. When looking at the stack throughput time, this KPI will improve slightly compared to the IPB time group selection strategy, but this is also to be expected.

For a less busy day, the results are very comparable. One thing that stood out is that lane selection prioritising filling the buffer to the maximum group size performed worse than other strategies. This is also expected as it will take longer for groups to form so pickers can start a picking round on a less busy day. So, on a less busy day, prioritising filling to the minimum group size after prioritising the deadline of the totes results in the best results.

Optimal configuration of the manual pick process

The fifth sub-question aims to optimise the configuration of the manual process. This includes different parameters as discussed in the second and third research questions.

For the values of the group size, it is concluded that the maximum group size should be as high as possible, and the minimum group size should be adjusted to a value that ensures a high enough picker productivity. A small minimum group size would mean pickers would be assigned a group even when there are only a few totes in the buffer, causing them to walk inefficient pick rounds. For less busy days, the minimum group size should not be too high to prevent pickers from waiting too long, and the lane selection strategy should not prioritise filling buffers to maximum groups.

For the number of pickers, this value should be closely considered when the number of orders that should be fulfilled on a day is known. Too few pickers will be unable to keep up with the number of totes coming in, too many pickers will be inefficient, and personnel costs will rise. Having a few extra pickers than in minimally needed is preferred as toilet breaks and other disruptions of pickers can be compensated for.

Impact of disturbances

The sixth sub-question gives insight in the impact of disturbances in the system and how fast they have to be restored.

The most important finding from these experiments are that the system can not operate on a peak day with a consolidation station missing. With enough pickers, missing one consolidation station for 90 to 120 minutes would be possible, and the system could recover. This means that if a consolidation station breaks on a busy day, it should be repaired within 90 to 120 minutes. Otherwise, the system cannot catch up with the delays anymore. The buffer would become full, causing blockages in the zone pick system before the studied system and causing a big customer impact.

Most effective strategies and optimal configuration of manual picking process

An answer to the main research question can be formulated from the answers to the sub-questions. For the strategy combination of the lane selection and the group formation, it becomes clear that using the deadline for the lane selection combined with the IPB time for the grouping strategy results in better system performance. To create the most robust combination that can handle both busy and less busy days selecting the ”deadline min groups” strategy would be best. The ”deadline max groups” strategy would be best for busy days when the strategies can be changed. As picker productivity increases, the other KPIs would also improve marginally compared to the ”deadline min groups” strategy on busy days.

Some things are important to improve the system’s performance for the configuration of the manual process. One thing to note is that if a consolidation station breaks on a busy day, it should be repaired within 90 minutes. Otherwise, the system cannot recover. It is important to look at the number of orders that are planned to be fulfilled on a day and to plan the number of pickers accordingly. Planning at least 2 extra pickers above the minimum is important to compensate for human variation. If possible, the minimum group size value should also be changed according to the number of orders to be fulfilled daily. On a busy day, this number should be a bit higher to ensure higher picker productivity, and on less busy days, it should be lower to ensure a shorter sojourn time of orders. The maximum group size should be as high as possible when using a strategy prioritising minimum group size.

Combining these findings results in the most optimal performance based on the experiments that are in this research.

References

Azadeh, K., Koster, R. D., & Roy, D. (2019). Robotized and automated warehouse systems: Review and recent developments. *Transportation Science*, *53*, 917–945. https://doi.org/10.1287/trsc.2018.0873

Balamurugan, B. (2015). The operation data analysis research based on the queuing theory of the warehouse optimization. *Journal of Chemical and Pharmaceutical Sciences*, *7*, 270–273.

Bansal, V., & Roy, D. (2021). Stochastic modeling of multiline orders in integrated storage-order picking system. *Naval Research Logistics*, *68*, 810–836. https://doi.org/10.1002/nav.21978

Bhaskaran, K., & Malmborg, C. (1989). Modelling the service process in a multi-address warehousing system. *Applied Mathematical Modelling*, *13*, 386–396. https://doi.org/10.1016/0307-904X(89)90149-2

Bruzzone, A. G., Briano, E., Bocca, E., & Massei, M. (2007). Evaluation of the impact of different human factor models on industrial and business processes. *Simulation Modelling Practice and Theory*, *15*, 199–218. https://doi.org/10.1016/j.simpat.2006.09.020

Chew, E., & Tang, L. (1999). Travel time analysis for general item location assignment in a rectangular warehouse. *European Journal of Operational Research*, *112*, 582–597. https://doi.org/10.1016/S0377-2217(97)00416-5

Cimini, C., Lagorio, A., Pirola, F., & Pinto, R. (2019). Exploring human factors in logistics 4.0: Empirical evidence from a case study [9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019]. *IFAC-PapersOnLine*, *52*, 2183–2188. https://doi.org/10.1016/j.ifacol.2019.11.529

Dewa, P. K., Pujawan, I. N., & Vanany, I. (2017). Human errors in warehouse operations: An improvement model. *International Journal of Logistics Systems and Management*, *27*, 298. https://doi.org/10.1504/IJLSM.2017.084468

Eben-Chaime, M., & Pliskin, N. (1996). An integrative model for automatic warehousing systems. *International Journal of Computer Integrated Manufacturing*, *9*, 286–292. https://doi.org/10.1080/095119296131580

Eben-Chaime, M., & Pliskin, N. (1997). Operations management of multiple machine automatic warehousing systems. *International Journal of Production Economics*, *51*, 83–98. https://doi.org/10.1016/S0925-5273(94)00062-X

Eder, M. (2023). Analytical approach of merging a different number of storage aisle under a fully sequenced order. *International Journal of Advanced Manufacturing Technology*, *124*, 487–496. https://doi.org/10.1007/s00170-022-10519-2

Goldratt, E. M. (1990). *Theory of constraints*. North River Croton-on-Hudson.

Grambo, P., Mullick, T., Furukawa, T., Matoba, M., & Nasu, Y. (2019). Automatic sorting-and-holding for stacking heterogeneous packages in logistic hubs. *IFAC-PapersOnLine*, *52*, 109–114. https://doi.org/10.1016/j.ifacol.2019.10.008

Herroelen, W., & Leus, R. (2001). *On the merits and pitfalls of critical chain scheduling*. http://www.pdinstitute.com;

Hulett, M., & Damodaran, P. (2014). Fork and join queuing networks to predict order picking times at a warehouse. *IIE Annual Conference and Expo 2014*, 1203–1212.

Kose, S. Y., & Kilincci, O. (2020). A multi-objective hybrid evolutionary approach for buffer allocation in open serial production lines. *Journal of Intelligent Manufacturing*, *31*, 33–51. https://doi.org/10.1007/s10845-018-1435-6

Magnanini, M. C., Terkaj, W., & Tolio, T. A. (2022). A novel algorithm for optimal buffer allocation in automated asynchronous unreliable lines. *International Journal of Advanced Manufacturing Technology*, *122*, 4405–4419. https://doi.org/10.1007/s00170-022-09867-w

Malde, A., Isik, T., Lockaby, R., Gantt, A., & Thumser, G. (2022). Optimal team formation and job assignment to optimize warehouse operations. *Proceedings - Winter Simulation Conference*, *2022-Decem*, 1772–1782. https://doi.org/10.1109/WSC57314.2022.10015444

Naeem, M., Dobbie, G., Weber, G., & Alam, S. (2010). R-meshjoin for near-real-time data warehousing. *International Conference on Information and Knowledge Management, Proceedings*, 53–60. https://doi.org/10.1145/1871940.1871952

Pan, J.-H., & Wu, M.-H. (2009). Throughput estimation in a picker-to-part warehouse with a multiple picker and blocking considerations. *Proceedings - 15th ISSAT International Conference on Reliability and Quality in Design*, 474–479.

Pandit, R., & Palekar, U. (1993). Response time considerations for optimal warehouse layout design. *Journal of engineering for industry*, *115*, 322–328.

Rahman, S. (1998). Theory of constraints: A review of the philosophy and its applications. *International Journal of Operations and Production Management*, *18*, 336–355. https://doi.org/10.1108/01443579810199720

Randhawa, S., & Shroff, R. (1995). Simulation-based design evaluation of unit load automated storage/retrieval systems. *Computers and Industrial Engineering*, *28*, 71–79. https://doi.org/10.1016/0360-8352(94)00027-K

Roy, D., Krishnamurthy, A., Heragu, S., & Malmborg, C. (2009). Impact of zones on throughput and cycle times in warehouses with autonomous vehicles. *2009 IEEE International Conference on Automation Science and Engineering, CASE 2009*, 449–454. https://doi.org/10.1109/COASE.2009.5234102

Rubrico, J. I. U., Ota, J., Higashi, T., & Tamura, H. (2006). Scheduling multiple agents for picking products in a warehouse [Cited by: 17]. *Proceedings - IEEE International Conference on Robotics and Automation*, *2006*, 1438–1443. https://doi.org/10.1109/ROBOT.2006.1641911

Sarker, B. R. (1984). Some comparative and design aspects of series production systems. *IIE Transactions (Institute of Industrial Engineers)*, *16*, 229–239. https://doi.org/10.1080/07408178408974689

Shaaban, S., & Romero-Silva, R. (2021). Performance of merging lines with uneven buffer capacity allocation: The effects of unreliability under different inventory-related costs. *Central European Journal of Operations Research*, *29*, 1253–1288. https://doi.org/10.1007/s10100-019-00670-9

Shang, S., Yu, Z., Jiao, K., Huang, Y., Guo, H., & Wang, G. (2022). Design of cross-platform information retrieval system of library based on digital twins. *Computational Intelligence and Neuroscience*, *2022*. https://doi.org/10.1155/2022/7999091

Smith, J. M. G., & Cruz, F. R. (2005). The buffer allocation problem for general finite buffer queueing networks. *IIE Transactions (Institute of Industrial Engineers)*, *37*, 343–365. https://doi.org/10.1080/07408170590916986

Winkelhaus, S., Grosse, E. H., & Morana, S. (2021). Towards a conceptualisation of order picking 4.0. *Computers & Industrial Engineering*, *159*, 107511. https://doi.org/10.1016/j.cie.2021.107511

Yu, M., & de Koster, R. (2009). The impact of order batching and picking area zoning on order picking system performance. *European Journal of Operational Research*, *198*, 480–490. https://doi.org/10.1016/j.ejor.2008.09.011

A

Model description

A.1. Technical model description

The simulation model built with Simpy is built in an object-oriented way. This means that the model is built up of different classes. In the ipb_system class, the environment is initialised. This is the main class used for the simulation environment, and this class contains all other model elements. When initialising an instance of this class, the instances of the buffers, totes, pickers, consolidation stations and cart pick instances are also initialised and the model will be run for the specified time.

Totes enter the system based on a distribution as explained in subsection A.2.1. After a tote enters the system, it will call a function from the cart pick class to be added to a buffer. In the add_to_buffer function the first check is whether the tote is close to its deadline and has to end up in a priority lane. A function is called to chose the buffer the tote will have to go into.

In the function to chose where the tote has to go into the first check is if the tote has to go into a priority lane. If it doesn't have to go into a priority lane it will go into on of the available buffer lanes based on the specified lane selection strategy. Code A.1 and Code A.1 show how the strategies work in the model for the base strategy and the deadline minimal groups strategy. For the base strategy, the tote is put into the lane with the least totes. so first it is checked if a lane is empty and next if the number of totes in the buffer is equal to the minimal number of totes of all buffer lanes.

```
if self.entry_strategy == 'base':
    min_val = min(self.buffer_sizes.values())
    for buff in self.buffers.values():
        if buff.size == 0:
            return buff
        if buff.size == min_val:
            return buff
    raise ExitStrategyException(self.entry_strategy)
```

Code A.1: Base strategy

for the deadline minimal groups strategy, it can be seen that first, it is checked if any lane contains ungrouped items with the same deadline. If there are none it is checked if there are buffer lanes with fewer totes than the minimal group size. If that condition is not satisfied, it is put into a lane with the least totes.

```
if self.entry_strategy == 'deadline_min':
    same_deadline = {i: 0 for i in self.buffers.values()}
    chosen = []
    chosen_size = []
    min_val = min(self.buffer_sizes.values())
    for buff in self.buffers.values():
        if buff.size < self.max_group_size and buff.not_assigned_size > 0:
            for item in buff.items_not_assigned:
                if item.frameloading_deadline == tote.frameloading_deadline:
                    same_deadline[buff] += 1
    max_val = max(same_deadline.values())
    if max_val > 0:
        return max(same_deadline, key=same_deadline.get)
```

```
    for buff in self.buffers.values():
        if buff.size < self.min_group_size:
            chosen.append(buff)
            chosen_size.append(buff.size)
    if len(chosen) > 0:
        return chosen[np.argmax(chosen_size)]
    for buff in self.buffers.values():
        if buff.size == 0:
            return buff
    for buff in self.buffers.values():
        if buff.size >= self.min_group_size and buff.size == min_val:
            return buff
    raise ExitStrategyException(self.entry_strategy)
```

Code A.2: Deadline minimal groups strategy

The pickers regularly check whether a lane has enough totes to form a group. One exception is when no picks can be done for a certain amount of time, and there are totes in the buffer the picker can pick for a smaller group. This ensures that totes can still be picked for at the end of the day, and the system can be emptied.

```
self.calculate_buffer_sizes()
time_since_last_pick = self.env.now - self.last_pick_timestamp
if self.check_group_available() or (time_since_last_pick > self.pick_treshold and max(
        self.buffer_sizes_no_pick.values()) > 0):
    picker = self.select_picker()
    chosen = self.choose_get_buffer()
    group = self.get_group(chosen)
    chosen.groups.append(group)
```

Code A.3: Check if group available

When there are enough totes to form a group or the totes in the IPB have been waiting for too long, the group selection strategy is called. In the group selection, the first check is whether there are totes in a priority buffer and if there are, these totes are selected first as they have priority over other totes. Otherwise, the group selection strategy is used. The group strategy IPB Time is shown in Code A.1

```
        if self.group_strategy == 'entry':
            entries = self.get_sojourn(self.buffer_items_no_pick)
            removed = []
            removed_entries = {}
            # removed_check = []
            for key, val in entries.items():  # For now, delete buffers with size <
                group_size
                if len(val) < self.min_group_size:
                    removed.append(key)
                if len(val) > self.min_group_size:
                    entries[key] = val[:max(self.max_group_size, self.buffer_sizes[key])]
            for key in removed:
                removed_entries[key] = entries[key]
                del entries[key]
            if len(entries) == 0:
                chosen_exception = []
                debug = False
                if debug:
                    print(
                        f'Exception in choose_get_buffer: No entries left in entries. Removed
                            entries: {removed_entries}')
                max_exception = max(removed_entries.values())
                for buff, vals in removed_entries.items():
                    if sum(vals) == sum(max_exception):
                        chosen_exception.append(self.buffers[buff])
                return chosen_exception[0]
            max_val = max(entries.values())
            for buff, vals in entries.items():
                if sum(vals) == sum(max_val):
                    chosen.append(self.buffers[buff])
            return chosen[0]
```

Code A.4: Group strategy IPB Time

After the picker has done the picking round, the code sets the picking_done attribute of all totes in the group to true so that a consolidation station can call them. Consolidation stations check regularly if a group is available. If a group is available, the consolidation station will call that group, the totes of that group will come to the consolidation station and the consolidation is done for the totes in that group. After consolidation, the totes exit the system and log to the output loggers.

A.2. Data

To show what input values are present in the model Table A.1 shows the input data points for the model.

Table A.1: Data points

Data type	Data point	Explanation
Order data	Arrivals	The number of orders and their characteristics is known in advance. Assumptions must be made to estimate when totes will enter the Manual pick area as the zonepick precedes the system studied in the model, and the fulfilment centre is not yet in operation.
	Products	Existing data shows what orderlines are in an order. Using some rules, products that will be in cart pick are known. Low OLS (sales) Glass products Low DQ (Decant quantity)
	Shipment/ Framenumber	Existing data have totes linked to shipment and frame numbers
Process times	Pick times	Data from manual fulfilment centres (FCs) can aid in estimating the pick times. What is the impact of #orderlines on the total pick time? What is the distribution of the variation in these times?
	Consolidation times	Data from the Automated FC can aid in estimating the consolidation times. How long does it take to consolidate N products?
	Conveyor times	The conveyor times can be taken from the actual system. Some future improvements could be: Reduce travel times when a buffer is filling up
Random events Events	Human error	Are there any random events that can have an impact on the system? e.g. A worker goes to the bathroom during pick round An accident happens during pick round. These events can be incorporated into the model and their effects can be studied.
	Equipment error	Are there any equipment malfunctions? e.g. Conveyor breaks down Consolidation station breaks down These events can be incorporated into the model and their effects can be studied.

A.2.1. Arrivals

For the arrival of totes, a dataset is available from Picnic with orders for one day of operation in Dordrecht. These totes are in order of tote id, stack id and shipment id, so in order to get a realistic arrival pattern in the system for the model, the totes should be spread according to a realistic distribution. After talking to experts within Picnic, it was agreed that looking at the distribution of the shipment ids of orders that are fulfilled in Picnic's automated fulfilment centre. Zone pick is combined with Goods to Person (GTP) picking in this fulfilment centre. The most realistic point to look at the spread of the shipment ids would be the timestamp of the last pick in zone pick.

In order to get this spread. SQL was used to combine tables from Picnic's data warehouse to obtain the timestamp of the last zonepick pick. These timestamps could then be plotted in a histogram to see the spread. This spread is given for 9 shipment ids in Figure A.1.

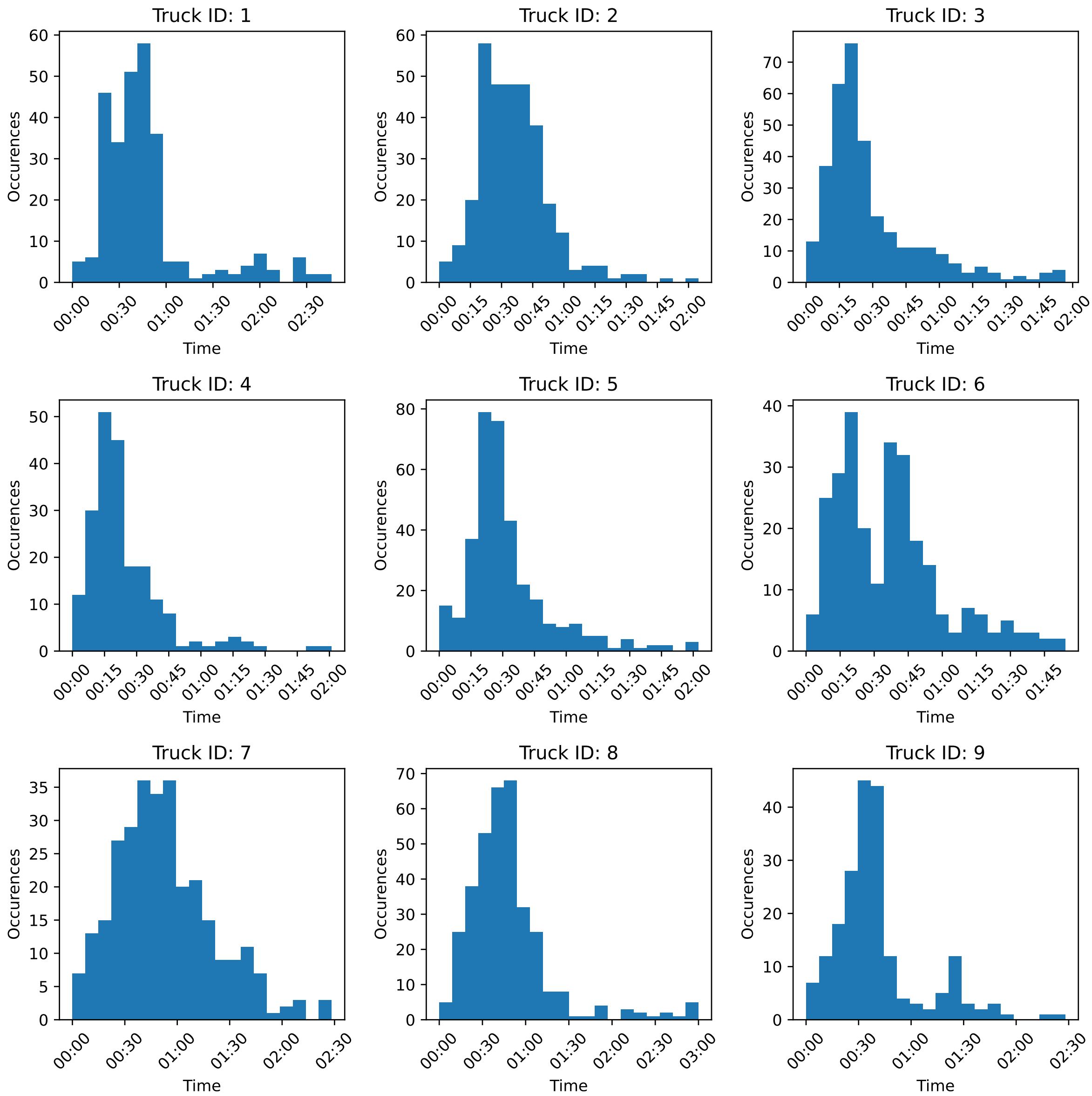

Figure A.1: Distribution of timestamp zone pick done in FCA

As can be seen from the figure most distributions look relatively the same. After looking at different distributions and plotting different values the distribution that resembled this distribution the closest was the lognormal distribution. This lognormal distribution with said parameters is given in Figure A.2.

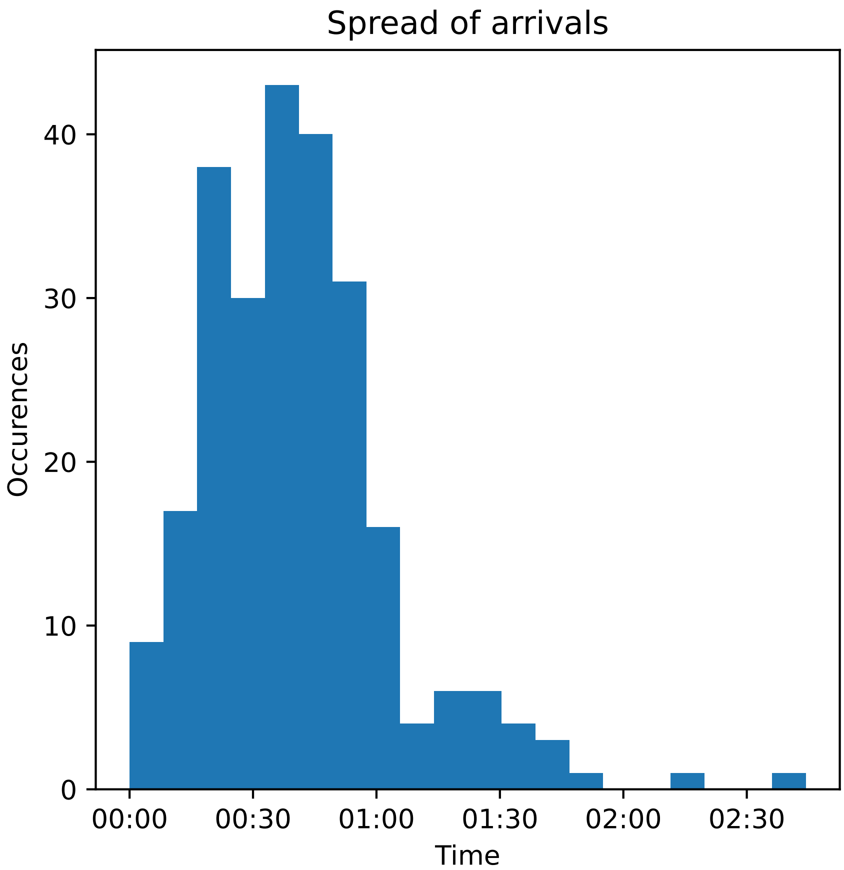

Figure A.2: Lognormal distribution with parameters [0.1,0.35]

B
Buffer strategies

This chapter explains the strategies for lane selection and group formation. The strategies will impact the model's outcome, and with the results, trade-offs will have to be made between different KPIs. Different strategies will impact the KPIs in their way, and finding the right strategy to maximise the system's performance is very important.

The model can use different strategies for lane selection and group formation. For lane selection, once a tote enters the system, it must decide which lane it will go. Once a tote has entered a lane, it will stay there until it has formed a group with other totes in that lane, and a picker will start a pick round for them. Totes have different properties, and lanes have different values based on which the totes can be added to a specific lane. The number of totes in a lane is important to remember, as the lanes in the studied system have a physical limit of 22 totes. Also, the number of totes in a lane is important as it can help create bigger groups for the pickers to start a pick round for. A pick round consists of a base time a worker needs to complete a circuit and an added time based on the number of picks that must be done (the number of orderlines of all orders combined). Because of this, picking for a bigger group results in pickers being more efficient. A downside to this is that if a pick round lasts longer, the buffer will contain more totes, as during the pick round, more totes will enter the system (next to the totes waiting in the buffer for their pick round to finish).

Next to how many totes are in the buffer lane, other aspects can also be used to select lanes for totes. As some KPIs need to minimize the shipment and stack throughput time, these tote properties can also be used for lane selection. If there are totes in a lane with the same stack id, it might be good to put the tote in that lane same goes for the shipment id.

Figure B.1 through Figure B.5 show the flow charts of how the allocation strategies work.

Figure B.6 through Figure B.8 show the flow charts of how the group strategies work.

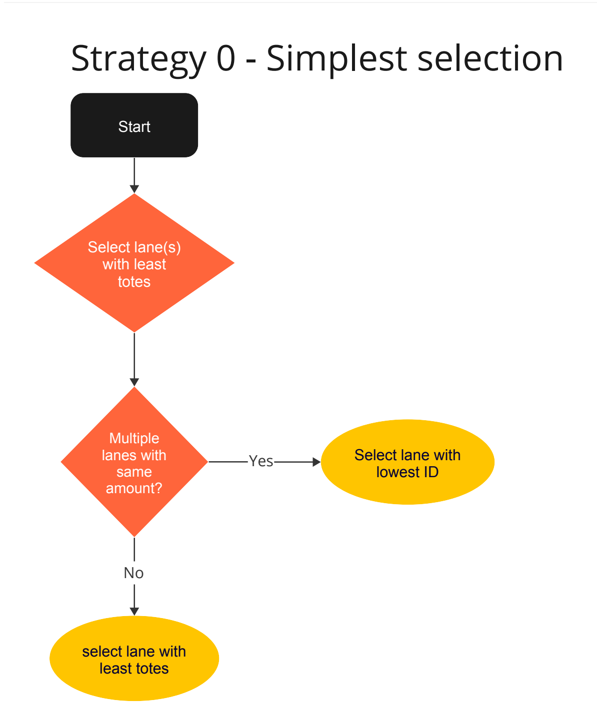

Figure B.1: Strategies: Base strategy

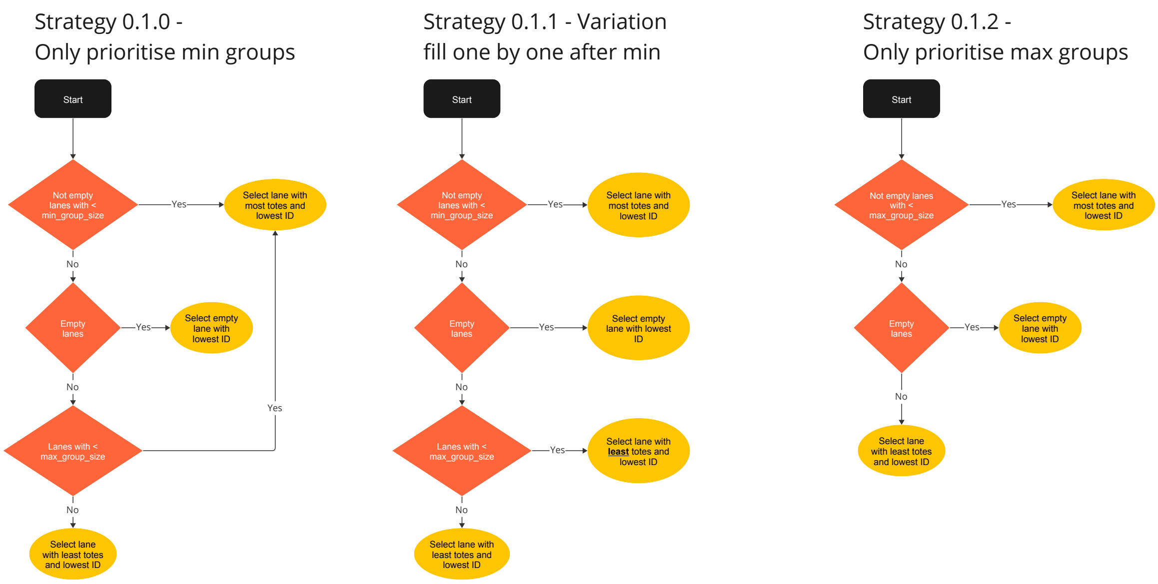

Figure B.2: Strategies: Grouping strategy

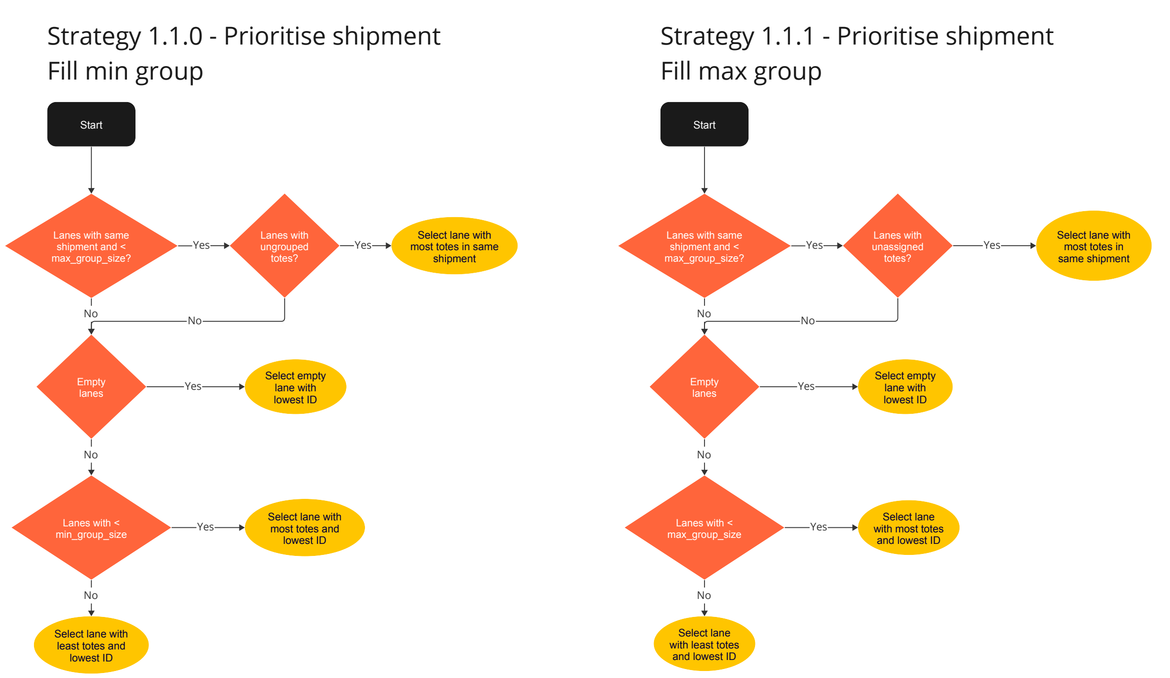

Figure B.3: Strategies: Shipment strategy

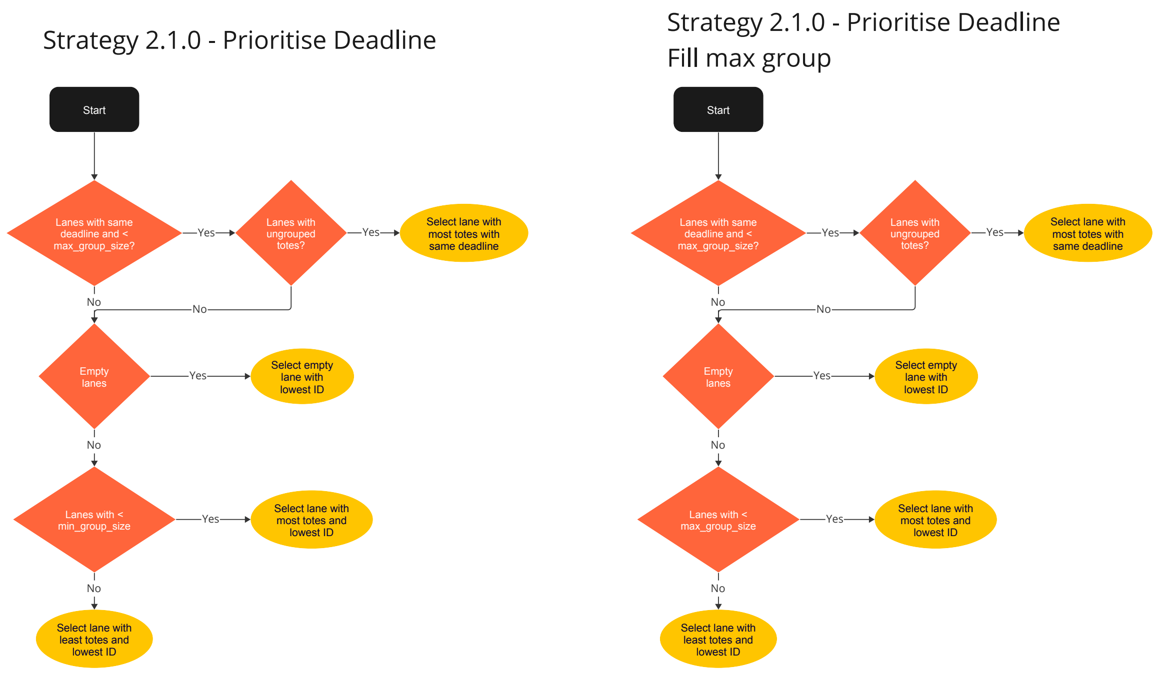

Figure B.4: Strategies: Stack strategy

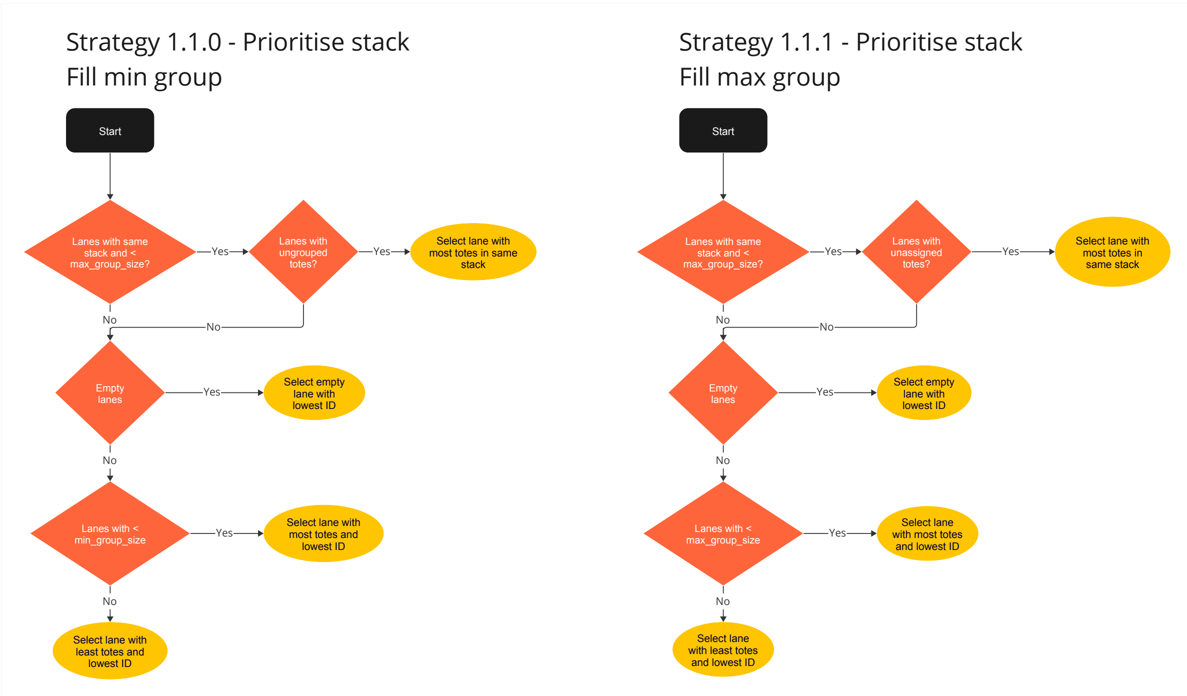

Figure B.5: Strategies: Deadline strategy

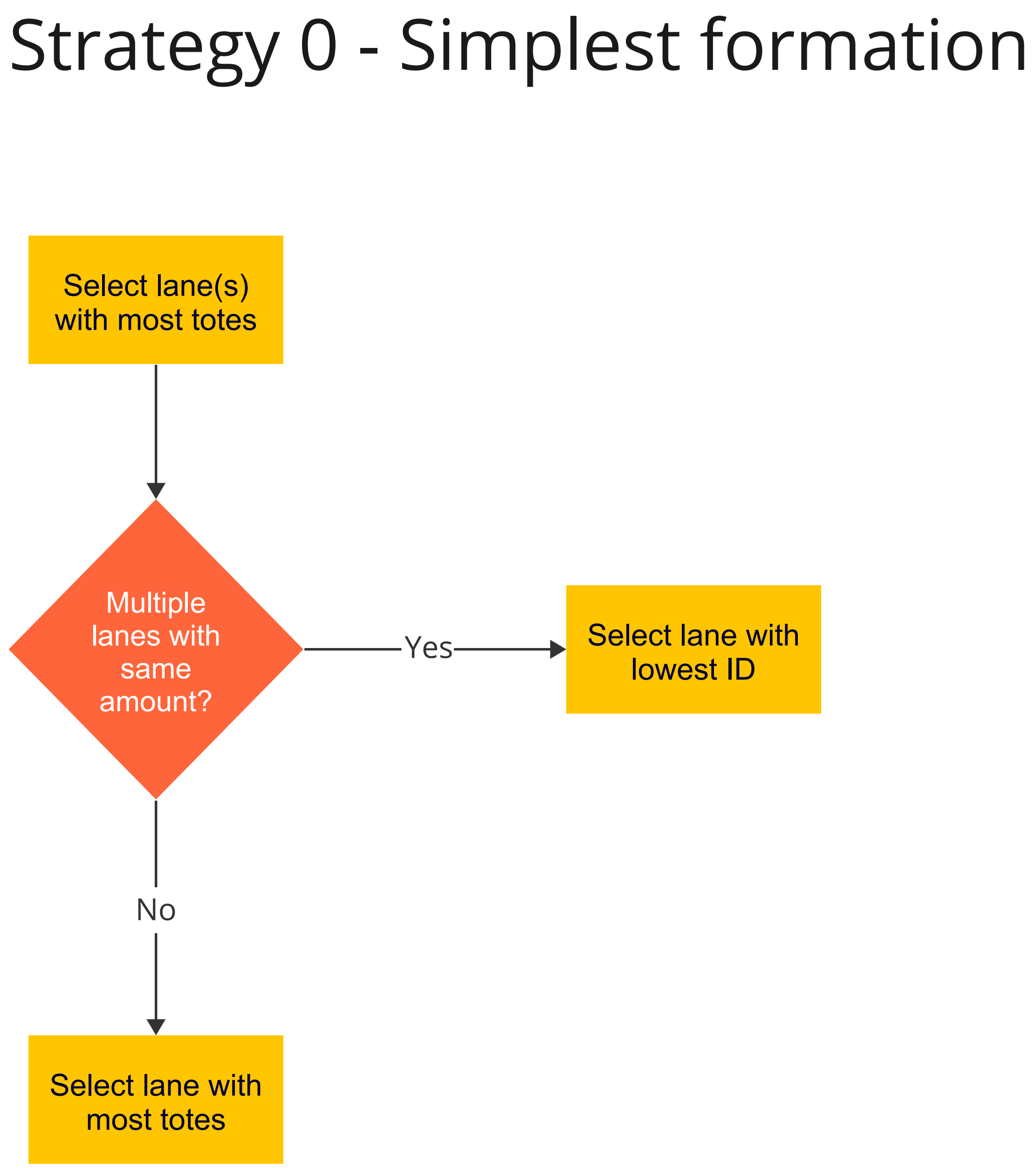

Figure B.6: Strategies: Group strategy

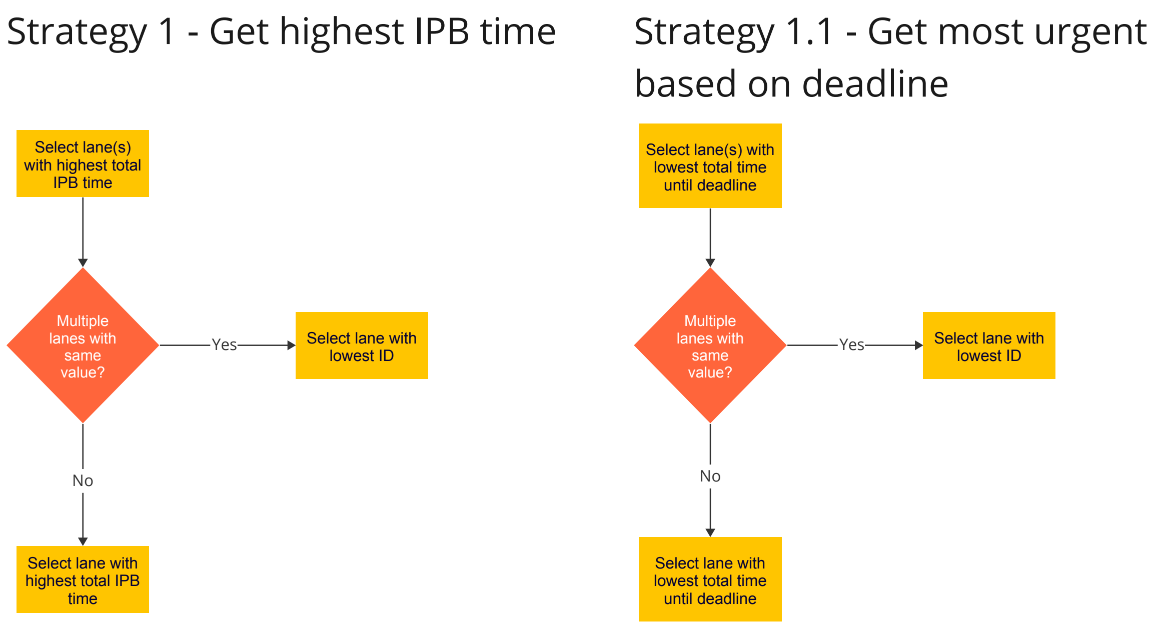

Figure B.7: Strategies: Group strategies

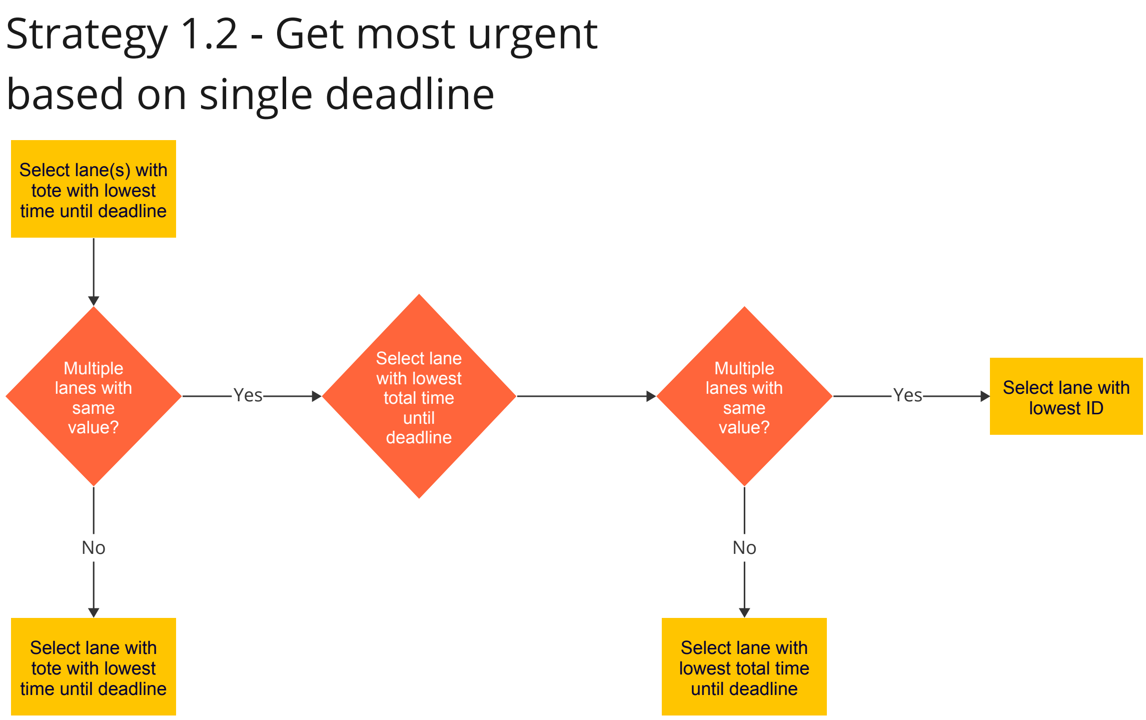

Figure B.8: Strategies: Group strategy

C

Validation and verififcation

C.1. Experiment values number of workers

Table C.1: Experiment values: Number workers

Ex#	# pickers	# consolidation
6	18	5
7	17	5
8	19	5
9	20	5
10	21	5
11	18	6
12	17	6
13	19	6
14	20	6
15	21	6

C.2. Verification

C.2.1. Priority totes

Another thing that should be checked is that totes that enter the system close to their deadline should be put into the priority lane. This can be checked by adding a breakpoint to the model entry when a tote is too close to its deadline and checking if that tote ends up in a priority lane. In the model, when a tote enters the system, its entry time is saved as an attribute. Then the tote is put into a buffer using a function. after that function, a tote that is too close to its deadline should be put into a priority lane. Figure C.1, Figure C.2 and Figure C.3 show how this was checked in the model. If

```
self.entry_time = self.env.now
deadline = cart_pick.date_to_model_time(self.frameloading_deadline)
time_to_deadline = deadline - self.env.now
if time_to_deadline < 30 * 60:
    pass
# Saving the buffer lane as an attribute of the tote while adding it to the buffer.
self.buffer_lane = cart_pick.add_to_buffer(self)
self.logger.tote_log(
```

Figure C.1: Priority validation 1.1

The figures show that tote 24 enters the system with 805 seconds until its deadline. The threshold is set to 30 minutes (1800 seconds), so this tote should end up in the priority lane. As we can see

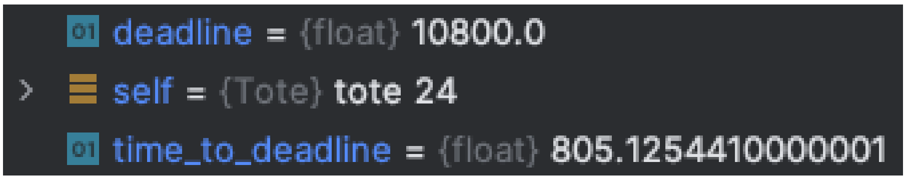

Figure C.2: Priority validation 1.2

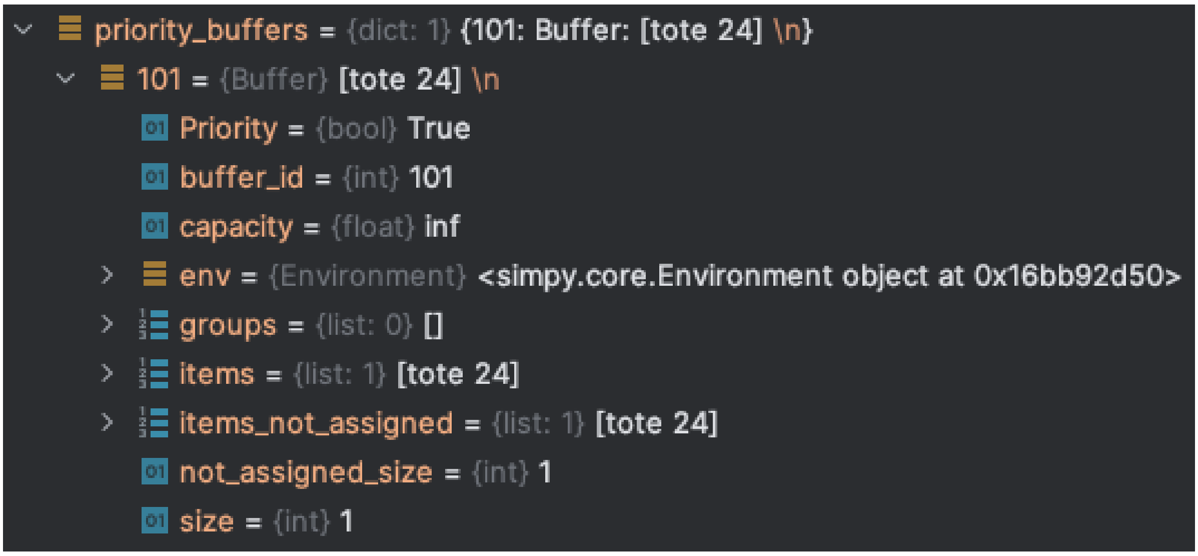

Figure C.3: Priority validation 1.3

from Figure C.3 this also happens; tote 24 is put into priority buffer 101. Also, the values for items not assigned, not assigned size and size are correctly updated.

To follow up on this, it can also be checked if the following picker that wants to form a group selects this tote from the priority lane. A breakpoint is set in the function to see if the next picker that requests a group gets tote 24 from the priority buffer.

As can be seen, two breakpoints are set in the function to see if the tote in the priority buffer is indeed selected first. This is the case, and when looking at Figure C.5, it is also clear that it is tote 24 from priority buffer 101. Also, it can be seen that the items not assigned and items not assigned size values have been updated because tote 24 has been assigned to a picker. This is also clear from the fact that tote 24 is now in a group in the buffer, as seen from the groups attribute.

We can also use tote 24 to check if the sojourn time seems right. For this priority tote, it should be quite low as it is selected with a priority in all processes. So when looking at this tote's number of products, pick time, and finally, its sojourn time, we can check if it looks plausible with the sojourn time.

```
def choose_get_buffer(self):   self: <cart_pick.CartPick object at 0x13a75e710>
    chosen = []   chosen: [tote 24] \n
    self.calculate_buffer_sizes()
    max_val = max(   max_val: 15
        self.buffer_sizes_no_pick.values())

    if any(self.priority_items_no_pick.values()):
        options = self.get_deadlines(self.priority_items_no_pick)   options: {101: [789.6476116852518]}
        min_val = np.inf   min_val: 789.6476116852518
        for buff, items in options.items():   buff: 101    items: [789.6476116852518]
            for item in items:   item: 789.6476116852518
                if item < min_val:
                    min_val = item
        for buff, items in options.items():
            if min_val in items:
                chosen = self.priority_buffers[buff]
        return chosen

    if self.group_strategy == 'base':
        for buff in self.buffers.values():
            if buff.not_assigned_size == max_val:
                chosen.append(buff)
        return chosen[0]
```

Figure C.4: Priority validation 1.4

```
chos = {Buffer} [tote 24] \n
    Priority = {bool} True
    buffer_id = {int} 101
    capacity = {float} inf
  > env = {Environment} <simpy.core.Environment object at 0x1753e3b90>
  > groups = {list: 1} [[tote 24]]
  > items = {list: 1} [tote 24]
  > items_not_assigned = {list: 0} []
    not_assigned_size = {int} 0
    size = {int} 1
```

Figure C.5: Priority validation 1.5

D
Experiments

Experiments will be defined to determine the best strategies to be used and the best configuration for the manual pick process. In this chapter, the experimental design will be explained. First, the setup of the experiments will be discussed. How many replications are needed per experiment and how the model run is set up are essential factors to be decided. Also, how the output is presented and what are acceptable values will be discussed in this chapter. Next, the experimental design per category will be presented. The categories for which experiments will be done are The strategies, the configuration of the manual pick process (how many workers per process, process times, and variation will be a part of this category), the possible disruptions in the system and lastly experiments on the design of the system hardware and layout.

D.1. Replications

One experiment is repeated across a different amount of replications to find how many replications will be needed to give accurate results. The optimal amount of replications gives reliable results. As the replications increase, the confidence interval decreases, and thus the results become more reliable. Table D.1 Shows the results of experiments with different numbers of replications.

As can be seen from the results, there is a difference in reliability between the different amounts of replications. If we look at the number of late totes, we can see that for 8 replications, the half width is 2.64. This means there is a 95% chance that the result will be within 2.64 below or above the average number of late totes. As this is an important KPI, higher reliability is preferred, and thus the number of replications should be higher. 32 replications give a reliable result for almost all KPIs. This number will be used for most experiments.

Table D.1: replications

8 replications	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time	Average	20,87125	20,3	22,88	0,70180867
	Maximum	72,92	48,67	102,95	16,5634418
IPB time	Average	16,11875	15,58	18,1	0,69378134
	Maximum	71,07	41,32	102,21	17,4650662
Number late	Total	10,375	6	16	2,64136876
stack throughput time	Average	115,6675	115,14	117,48	0,64768309
	Maximum	164,16625	156,94	179,34	7,52756073
picker UPH	Average	499,13625	498,37	499,73	0,44449584
	Minimum	466,6225	460,08	477,89	4,64664728
	Maximum	531,51	523,86	545,74	6,17607147
buffer items	Maximum	16,5	16	18	0,63197241

16 replications	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time	Average	20,674375	20,3	22,88	0,32625784
	Maximum	77,099375	48,67	106,39	10,075104
IPB time	Average	15,913125	15,53	18,1	0,32445861
	Maximum	75,864375	41,32	105,99	10,464072
Number late	Total	10,125	6	16	1,61478698
stack throughput time	Average	115,5375	115,11	117,48	0,3006874
	Maximum	166,525	156,52	181,49	4,65336369
picker UPH	Average	499,21125	498,37	499,97	0,25943978
	Minimum	464,604375	447,9	477,89	4,00445615
	Maximum	530,715625	515,98	545,74	4,29073772
buffer items	Maximum	16,1875	15	18	0,34908248

32 replications	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time	Average	20,5875	20,29	22,88	0,16309167
	Maximum	77,34875	48,67	106,39	6,83409081
IPB time	Average	15,8328125	15,52	18,1	0,16162273
	Maximum	76,185625	41,32	105,99	7,02662624
Number late	Total	9,4375	5	17	1,11760353
stack throughput time	Average	115,482188	114,95	117,48	0,15973202
	Maximum	164,93625	156,52	192,54	3,27715428
picker UPH	Average	499,435	498,37	500,98	0,20358711
	Minimum	464,415313	447,9	479,31	2,90462505
	Maximum	529,329688	515,98	550,62	3,05070284
buffer items	Maximum	16,3125	15	18	0,21354301

64 replications	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time	Average	20,5351563	20,09	22,88	0,08610039
	Maximum	77,0053125	46,21	106,39	4,42431668
IPB time	Average	15,7784375	15,36	18,1	0,08550387
	Maximum	76,0025	41,32	105,99	4,52752098
Number late	Total	9,265625	4	18	0,83054836
stack throughput time	Average	115,47	114,82	117,48	0,0891388
	Maximum	165,042031	156,52	200,7	2,71581698
picker UPH	Average	499,505938	498,37	500,98	0,13831291
	Minimum	464,160625	445	479,31	1,99426551
	Maximum	530,335938	515,98	557,17	2,08970365
buffer items	Maximum	16,34375	15	18	0,14227321

D.2. Strategies

For the strategies, it is important to vary the lane selection and group formation strategies and find if any combination will give superior results and if any specific strategy outperforms others. the strategies are extensively explained in Appendix B.

For the strategies a full factorial design will be analysed, combining all lane selection strategies with all group formation strategies. As the strategies are an important aspect of the research, finding the best combination of strategies is very important. It may be difficult to estimate possible synergies between different strategies so analysing all combinations eliminates missing possible good strategy combinations. The design is displayed in Table D.2. The combination that gives the best and most consistent results will also be used in the other experiments. After all the separate experiments have been done, different experiments with other strategies will be performed.

Table D.2: Experimental design: Strategies

Ex#	Lane selection strategy	Group formation strategy
S1	Base	Base
S2	Min groups	Base
S3	Min one by one	Base
S4	Max groups	Base
S5	Shipment Min group	Base
S6	Shipment Max group	Base
S7	Stack Min group	Base
S8	Stack Max group	Base
S9	Deadline Min group	Base
S10	Deadline Max group	Base
S11	Base	IPB Time
S12	Min groups	IPB Time
S13	Min one by one	IPB Time
S14	Max groups	IPB Time
S15	Shipment Min group	IPB Time
S16	Shipment Max group	IPB Time
S17	Stack Min group	IPB Time
S18	Stack Max group	IPB Time
S19	Deadline Min group	IPB Time
S20	Deadline Max group	IPB Time
S21	Base	Total deadline
S22	Min groups	Total deadline
S23	Min one by one	Total deadline
S24	Max groups	Total deadline
S25	Shipment Min group	Total deadline
S26	Shipment Max group	Total deadline
S27	Stack Min group	Total deadline
S28	Stack Max group	Total deadline
S29	Deadline Min group	Total deadline
S30	Deadline Max group	Total deadline
S31	Base	Single deadline
S32	Min groups	Single deadline
S33	Min one by one	Single deadline
S34	Max groups	Single deadline
S35	Shipment Min group	Single deadline
S36	Shipment Max group	Single deadline
S37	Stack Min group	Single deadline
S38	Stack Max group	Single deadline
S39	Deadline Min group	Single deadline
v40	Deadline Max group	Single deadline

D.3. Cart pick

The experiments around the manual pick processes aim to find the best workforce configuration, as well as analyse the impact of different process times and different variations in these process times. The first experiment will analyse the number of workers present, and the second part of the experiments will focus on the process times and variation.

The first experiments focus on the number of workers in manual processes. The design is given in Table D.3

Table D.3: Experimental design: Number workers

Ex#	# pickers	# consolidation
1	17	4
2	18	4
3	19	4
4	20	4
5	21	4
6	17	5
7	18	5
8	19	5
9	20	5
10	21	5
11	17	6
12	18	6
13	19	6
14	20	6
15	21	6

To see the impact of the pick times on the output of the model the pick times will be varied across different values. The pick times as in the base input are the closest to the real data as it was gathered from manual FCs of Picnic. But to investigate the impact of variation in the pick times, these experiments will be used. The design is displayed in Table D.4.

Table D.4: Experimental design: pick times

Ex#	Pick times	Value
1	base	250
	mean	14.4
	min	12
	max	18
2	base	275
	mean	14.4
	min	12
	max	18
3	base	250
	mean	14.4
	min	13
	max	20
4	base	250
	mean	18
	min	15
	max	23

These pick times will allow one to explore the impact of different pick times on the output of the model. Later on, these pick times can be combined with other experiments to see if they also have an effect on different configurations of other input values.

In order to see the impact of different minimum group sizes and maximum group sizes, these input values will also be experimented with. During the design of the model, it was found that the group size had a great impact on different KPIs and therefore should be thoroughly experimented with. Also for the group sizes it will be important to see what the combination with other experiments will bring. For example, the strategies that all look at the group size will be greatly impacted by the values for minimum and maximum group size. To understand the isolated effect of the group size, different values will be tested for the minimum group strategy and the maximum group strategy. the design is given in Table D.5.

Table D.5: Experimental design: group size

Ex#	Min group size	Max group size	Strategy
1	3	15	Deadline min
2	5	15	Deadline min
3	7	15	Deadline min
4	9	15	Deadline min
5	11	15	Deadline min
6	5	11	Deadline min
7	5	13	Deadline min
8	5	15	Deadline min
9	5	17	Deadline min
10	5	19	Deadline min
11	3	15	Deadline max
12	5	15	Deadline max
13	7	15	Deadline max
14	9	15	Deadline max
15	11	15	Deadline max
16	5	11	Deadline max
17	5	13	Deadline max
18	5	15	Deadline max
19	5	17	Deadline max
20	5	19	Deadline max

to see the effect of the group size on the strategies, all lane selection strategies have been combined with small and big values for min and max group sizes to understand the effect. The experimental design is shown in Table D.6. This experiment will be run with the full and small dataset to see if a different group size should be used on peak days and less busy days.

Table D.6: Experimental design: group size and strategy

Ex#	Lane selection	Min group size	Max group size
G1	Base	3	12
G2	Min groups	3	12
G3	Min one by one	3	12
G4	Max groups	3	12
G5	Shipment Min group	3	12
G6	Shipment Max group	3	12
G7	Stack Min group	3	12
G8	Stack Max group	3	12
G9	Deadline Min group	3	12
G10	Deadline Max group	3	12
G11	Base	3	19
G12	Min groups	3	19
G13	Min one by one	3	19
G14	Max groups	3	19
G15	Shipment Min group	3	19
G16	Shipment Max group	3	19
G17	Stack Min group	3	19
G18	Stack Max group	3	19
G19	Deadline Min group	3	19
G20	Deadline Max group	3	19
G21	Base	10	12
G22	Min groups	10	12
G23	Min one by one	10	12
G24	Max groups	10	12
G25	Shipment Min group	10	12
G26	Shipment Max group	10	12
G27	Stack Min group	10	12
G28	Stack Max group	10	12
G29	Deadline Min group	10	12
G30	Deadline Max group	10	12
G31	Base	10	19
G32	Min groups	10	19
G33	Min one by one	10	19
G34	Max groups	10	19
G35	Shipment Min group	10	19
G36	Shipment Max group	10	19
G37	Stack Min group	10	19
G38	Stack Max group	10	19
G39	Deadline Min group	10	19
G40	Deadline Max group	10	19

D.4. System design

As the system that has been studied is already built, the dimensions are already known. In order to see if any other design of the system will have a significant impact on the performance of the system this section will experiment with different designs of the system. This will help future applications of buffer systems and aid to show the bottlenecks within the hardware of the system. The design is given in Table D.7.

Table D.7: Experimental design: Configurations

Ex#	Buffer lanes	Priority lanes	Max group size
1	19	1	15
2	21	1	15
3	23	1	15
4	25	1	15
5	19	1	20
6	21	1	20
7	23	1	20
8	25	1	20
9	19	1	25
10	21	1	25
11	23	1	25
12	25	1	25

D.5. Consolidation outage

As seen from the experiment in Section D.3 the system can't work on a peak day with one consolidation station missing. Because consolidation stations can break it is a good idea to test for how long a consolidation station can be broken before the system can't catch up with the disturbance. If it becomes clear what time still can be catched up with it is an interesting result for Picnic as they would know how fast a maintenance engineer has to be on site and how fast it has to be repaired. The design is given in Table D.8.

Table D.8: Experimental design: Consolidation outage

Ex#	Consolidation outage	Pickers
1	0	18
2	30	18
3	60	18
4	90	18
5	120	18
6	150	18
7	180	18
8	210	18
9	0	22
10	30	22
11	60	22
12	90	22
13	120	22
14	150	22
15	180	22
16	210	22

E

Results

intro to chapter

E.1. Strategies

Some interesting results can be found from the results of the full factorial experimental design of the strategies. The first thing that immediately catches the eye is the base strategy for lane selection strategy creates a very high uncertainty in the picker productivity compared to the other strategies. The average sojourn time and average IPB time are very similar, which is to be expected as the most considerate amount of time that a tote is in the system, it will be in the IPB. After it leaves the IPB only consolidation is left, which is very stable with enough workers this will count for all experiments. Because of this it is decided that only the total sojourn time of future experiments will be shown.

When looking at the most important KPI, the number of late totes when leaving the system, some strategy combinations are very bad but most of the combinations are in a lower range. The combinations that stand out the most are the combinations from experiments 15, 16, 19 and 20. They have low numbers of late totes and also their values for sojourn time and stack throughput time are low. These strategy combinations are:

shipment min group & IPB Time
shipment max group & IPB Time
deadline min group & IPB Time
deadline min group & IPB Time

As the shipment is closely related to a tote's deadline, it makes sense that these strategies are closely correlated. Interestingly, for the group formation strategy, the best combination with these strategies is grouping based on IPB time and not on the deadline of the totes in the IPB or the deadline of a single tote. It seems when taking the deadline into account for the lane selection, taking it into account for the group formation as well doesn't add any value and sometimes even makes it worse.

From the average stack throughput time, it is very clear that the last two grouping strategies improve this KPI more than the rest. which is to be expected as totes that are in the same stack also have the same deadline. So grouping based on the deadline also groups these totes together.

Another thing that catches attention is that the minimum group strategy for lane selection and the base strategy for group formation is very bad for some KPIs. This makes sense because after filling every lane with minimum group size, the lanes are filled to maximum group size and the group strategy groups totes from the lane with the most totes. This will most likely result in the last lanes not being picked for group formation very often. Since no strategy considers the deadline, these totes will have a low priority in this strategy combination.

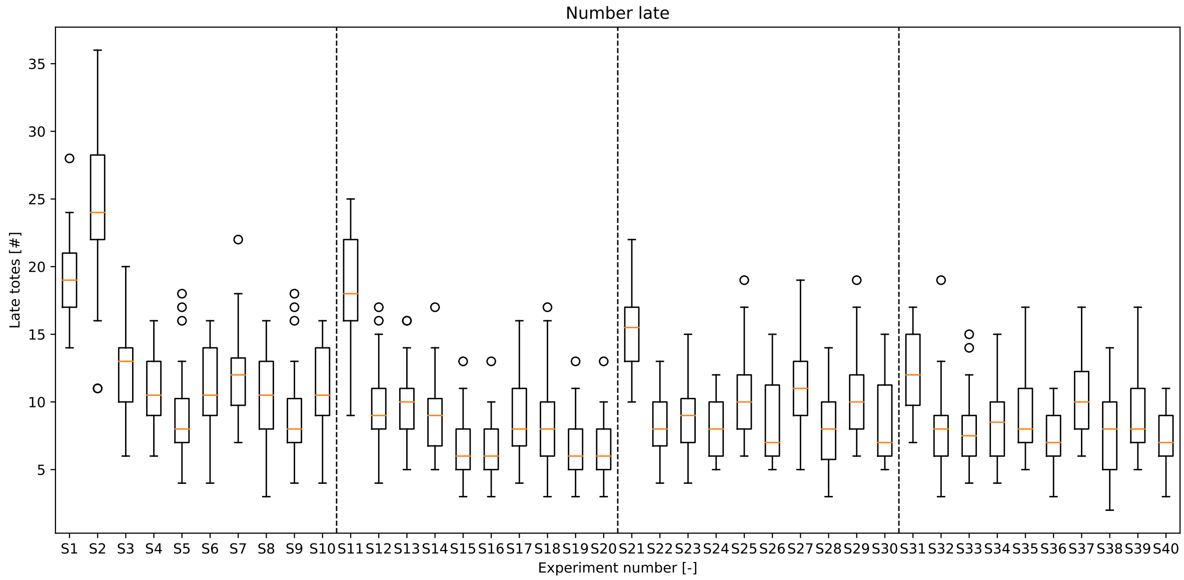

Figure E.1: Results Strategies: Number late

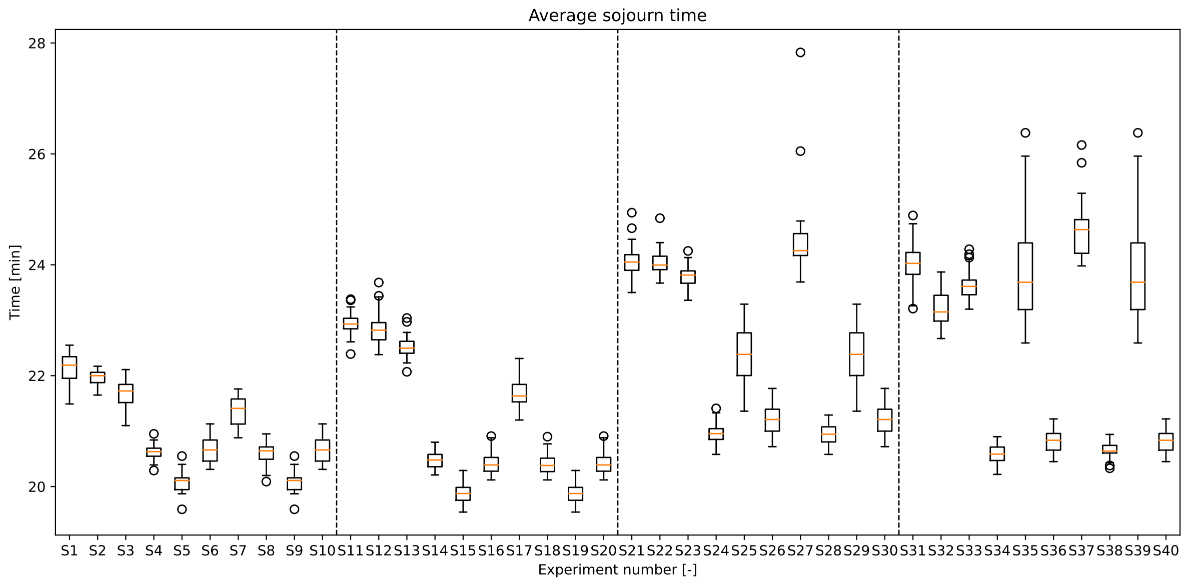

Figure E.2: Results Strategies: Sojourn time

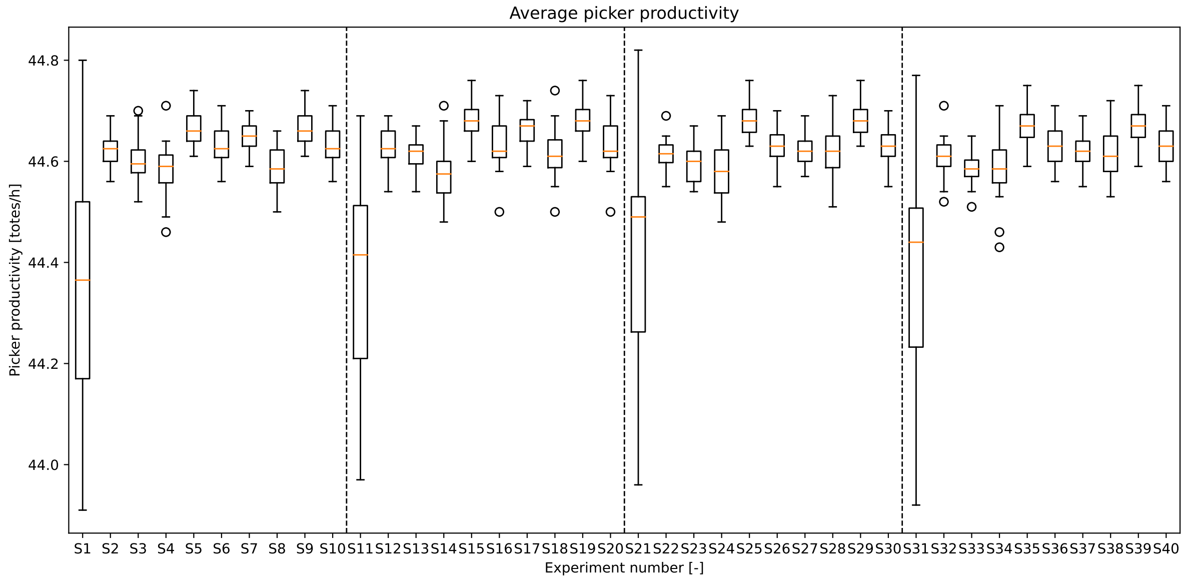

Figure E.3: Results Strategies: Picker productivity

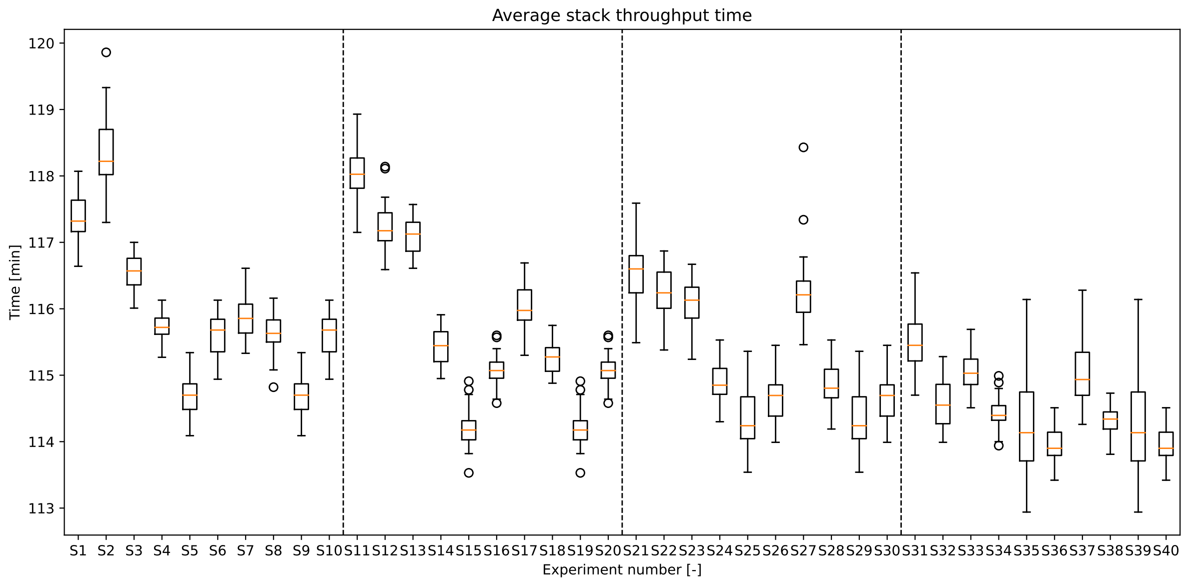

Figure E.4: Results Strategies: Stack throughput time

E.2. workers

One thing that is very clear from the experiment results for the number of workers is that with the full dataset, the system cannot operate stably when one consolidation station is missing. This is to be expected as the full dataset is based on the design capacity of the facility in Dordrecht and taking out one-sixth of the consolidation capacity for one day is detrimental. This does allow for some interesting experiments for the consolidation capacity. A future experiment will test different times when a consolidation station breaks with the full dataset and see how long it can be missed before the system cannot restore itself. This will allow for a conclusion on how long maintenance operators must fix a broken consolidation station if one breaks on a peak day.

From Figure E.9 on, the results are shown for 6 consolidation stations, so it is clearer what the results are for the number of pickers.

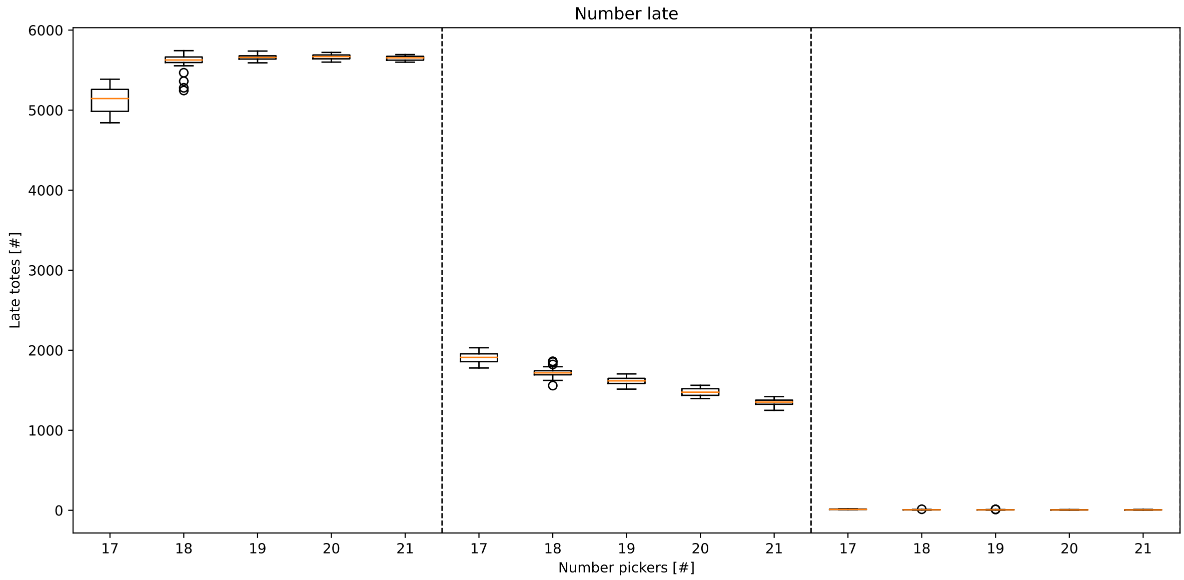

Figure E.5: Results Number of workers: Number late

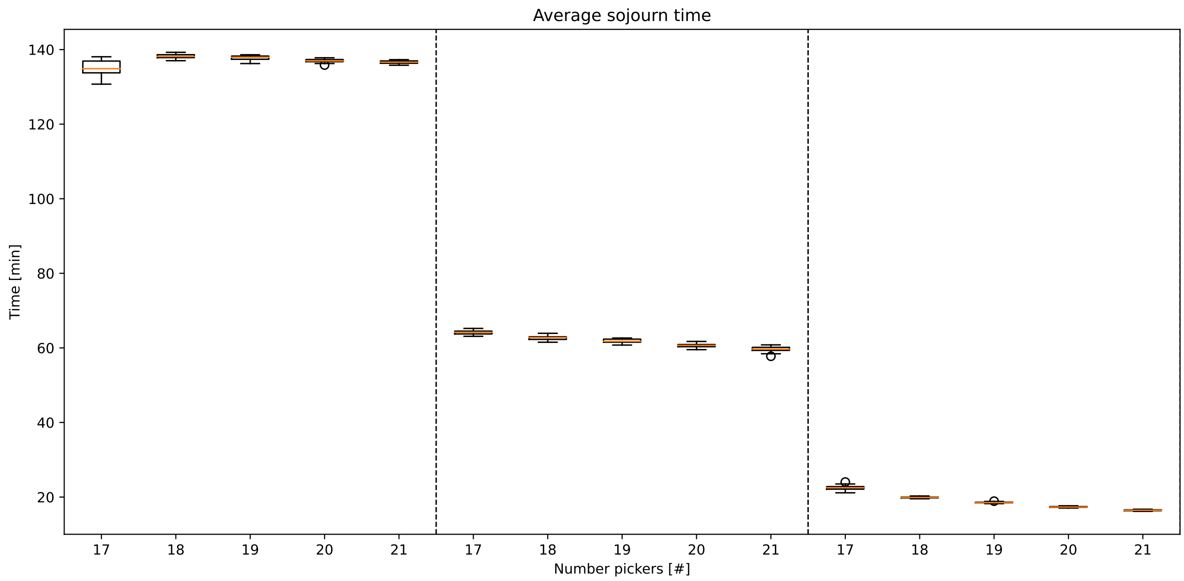

Figure E.6: Results Number of workers: Sojourn time

As can clearly be seen from Figure E.9 to Figure E.12. 17 pickers is not enough as the number

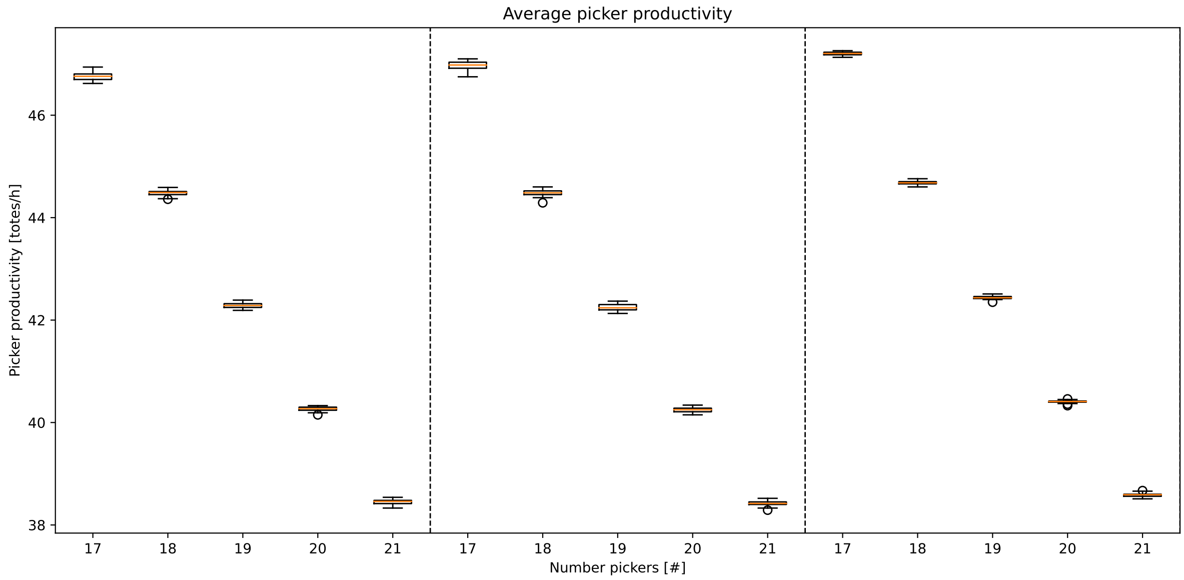

Figure E.7: Results Number of workers: Picker productivity

of late totes stretches too high. One interesting thing is that from 18 pickers on, the number of late totes doesn't go down as much anymore. The confidence interval does go down a bit, as can also be seen from the raw output in Section F.2. so for the number of pickers, it might be a good idea to have more than the threshold amount which is 18 pickers in the model. Any less will cause more late totes. Having more pickers does improve the other KPIs but won't have a very big impact on the number of late totes. One thing to note is that in the model the pickers are always working. This is done on purpose to make comparing experiments more reliable. In real life, humans will take breaks etc. This should be compensated by having more pickers than the minimum required amount.

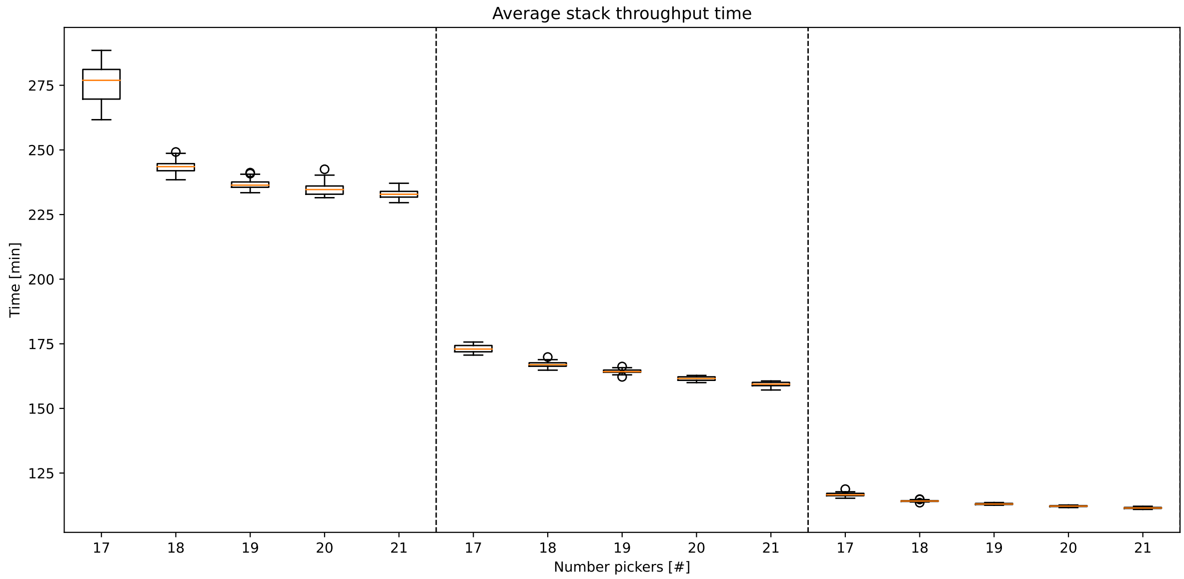

Figure E.8: Results Number of workers: Stack throughput time

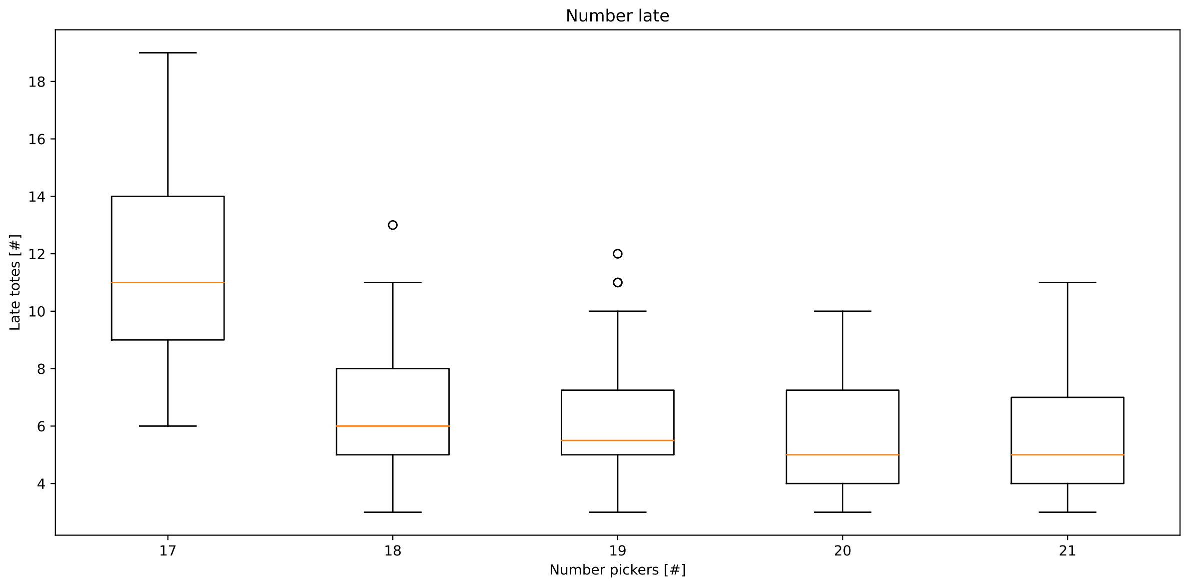

Figure E.9: Results Number of pickers: Number late

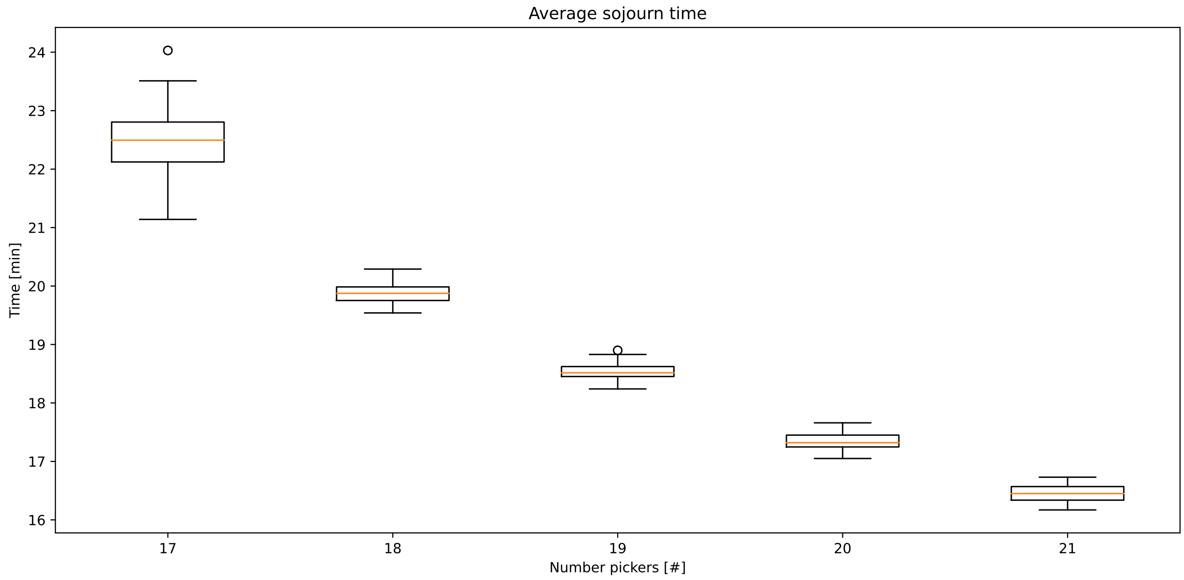

Figure E.10: Results Number of pickers: Sojourn time

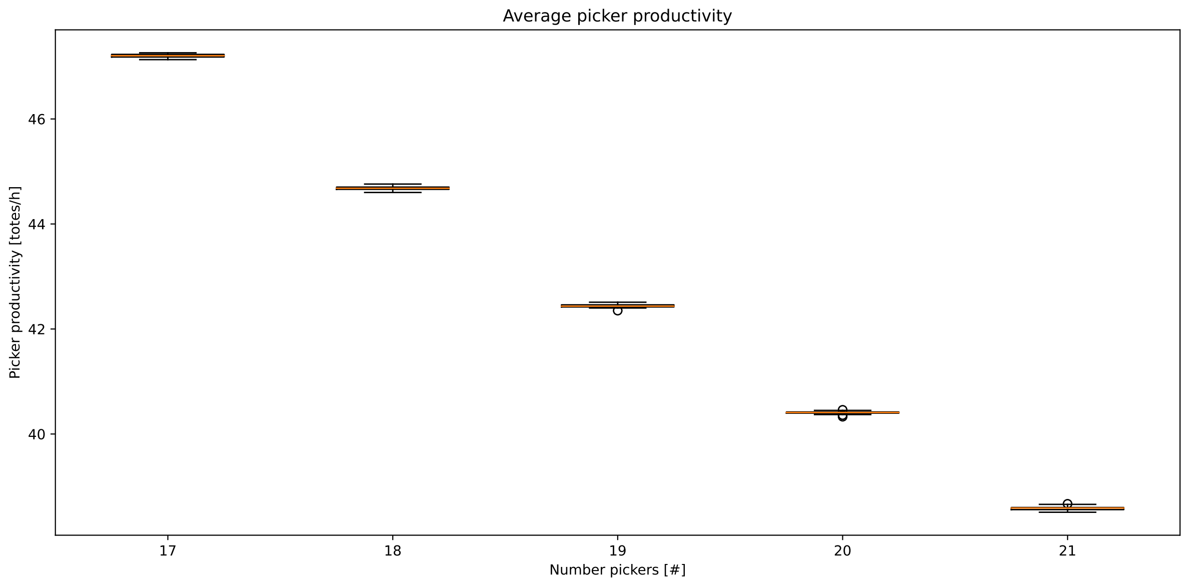

Figure E.11: Results Number of pickers: Picker productivity

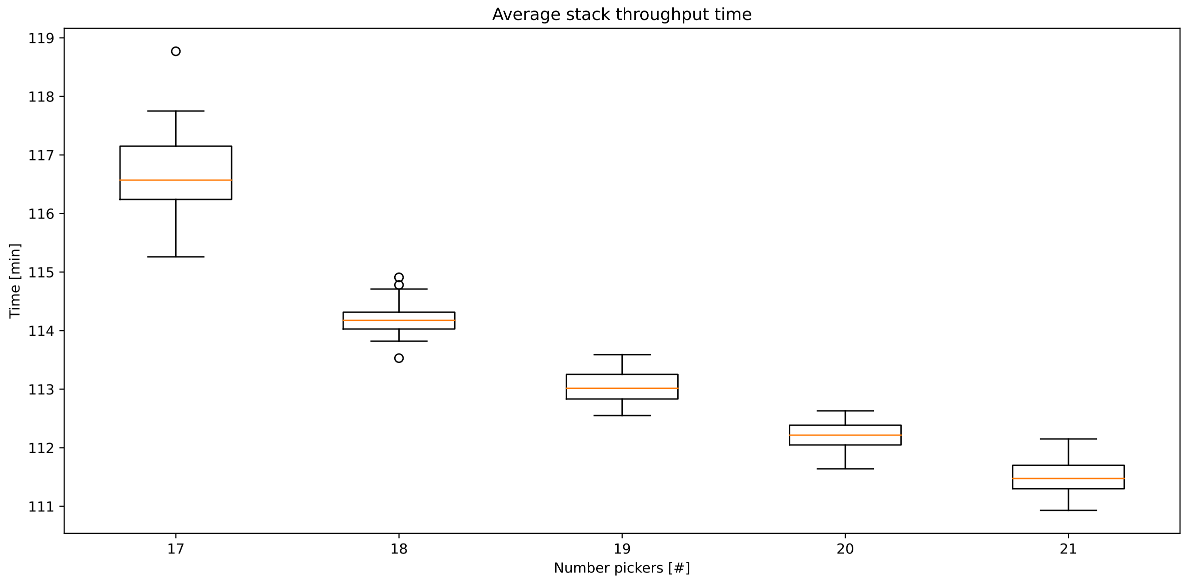

Figure E.12: Results Number of pickers: Stack throughput time

E.3. Pick times

For the pick times some basic variations are chosen and later on the pick time can be more of an added value when experimenting together with more optimal configurations of other aspects of the system. One thing that can be seen from these results is that there is a clear boundary in how high the pick times can become before the system fails. These results are for a system with 18 pickers which is on the edge of what the system can handle so this makes sense.

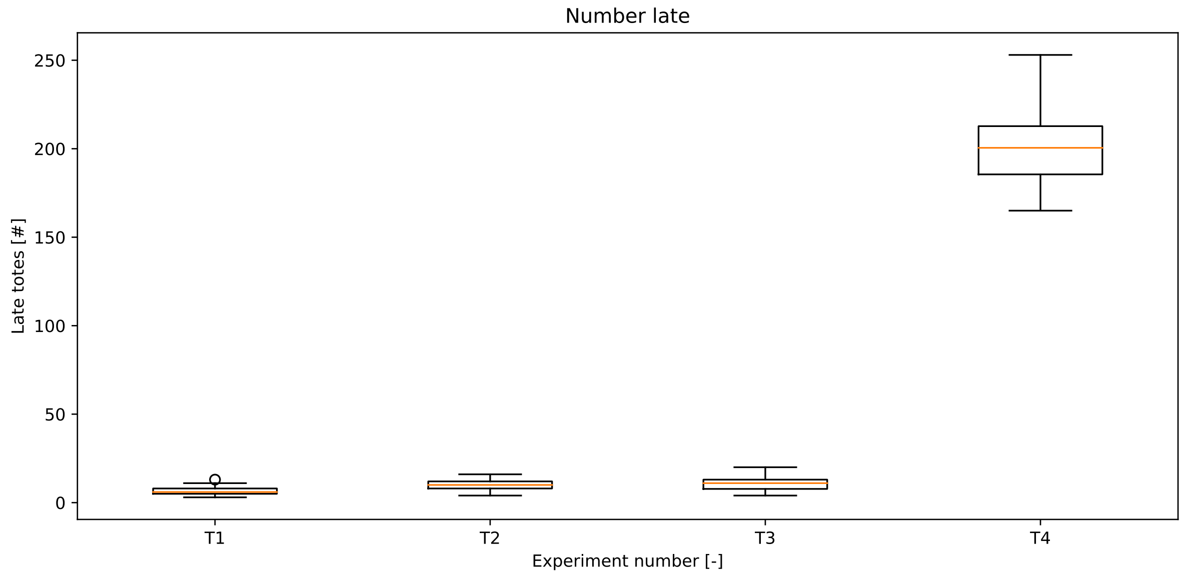

Figure E.13: Results Pick times: Number late

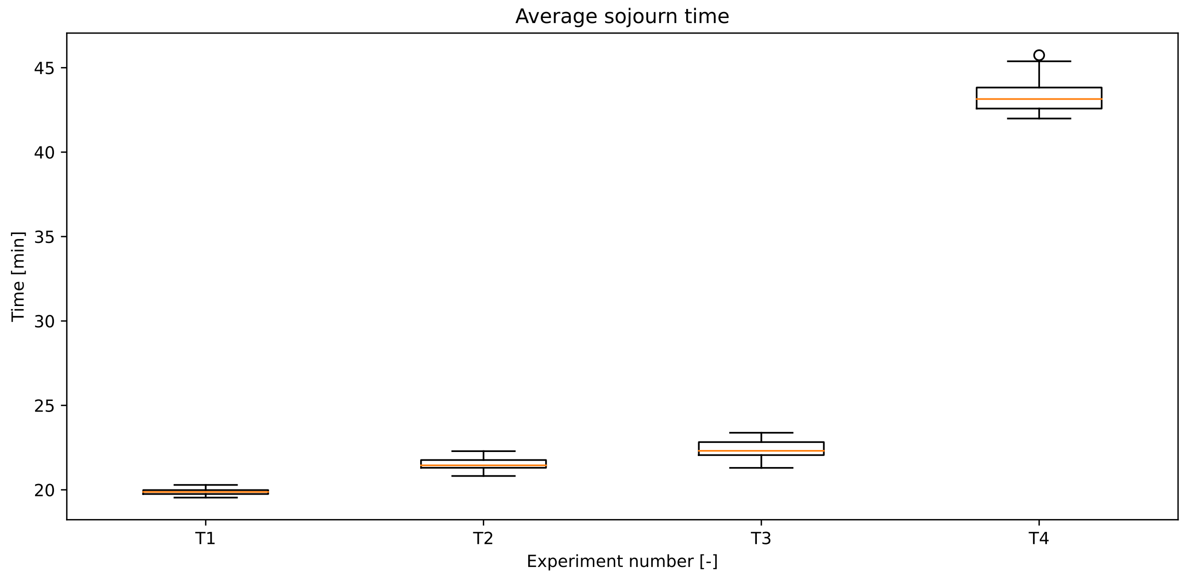

Figure E.14: Results Pick times: Sojourn time

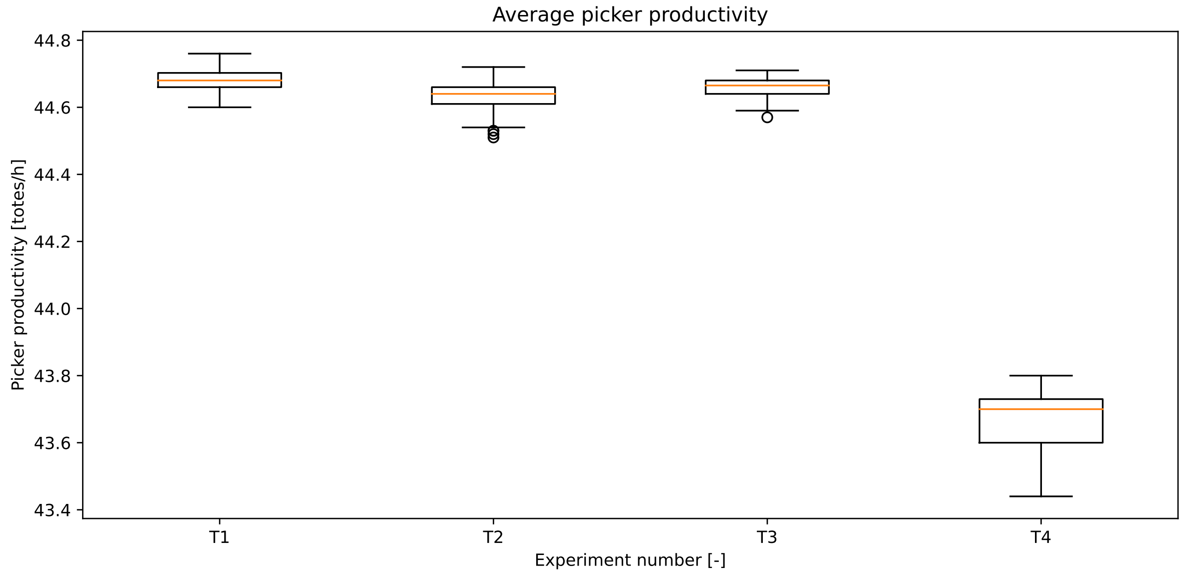

Figure E.15: Results Pick times: Picker productivity

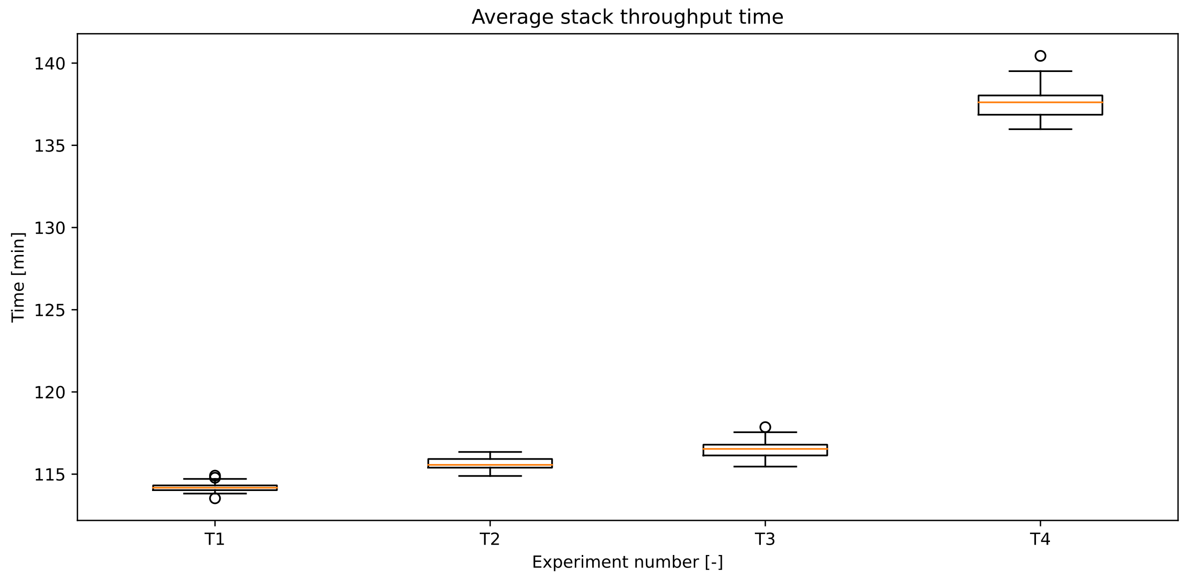

Figure E.16: Results Pick times: Stack throughput time

E.4. Group size

For the group size, it is interesting to see that most experiments give good and comparable results, but one combination clearly underperforms in comparison to the rest. A small max group size is not efficient to use on a busy day, as this is also with the full dataset. So on a peak day, using a larger max group size is best. Experiments with the small dataset should point out if this also goes for a less busy day and if Picnic should consider changing the group sizes based on the number of orders per day.

One interesting thing that can be seen in the picker UPH is that the average picker UPH goes up for the first experiments. This makes sense because as the minimum group size increases the number of totes that pickers do a pick round for also increases. The pickers wait until they can do a more efficient pick round. So, on a busy day it could make more sense to increase the minimum group size in order to improve the picker efficiency while maintaining other KPIs.

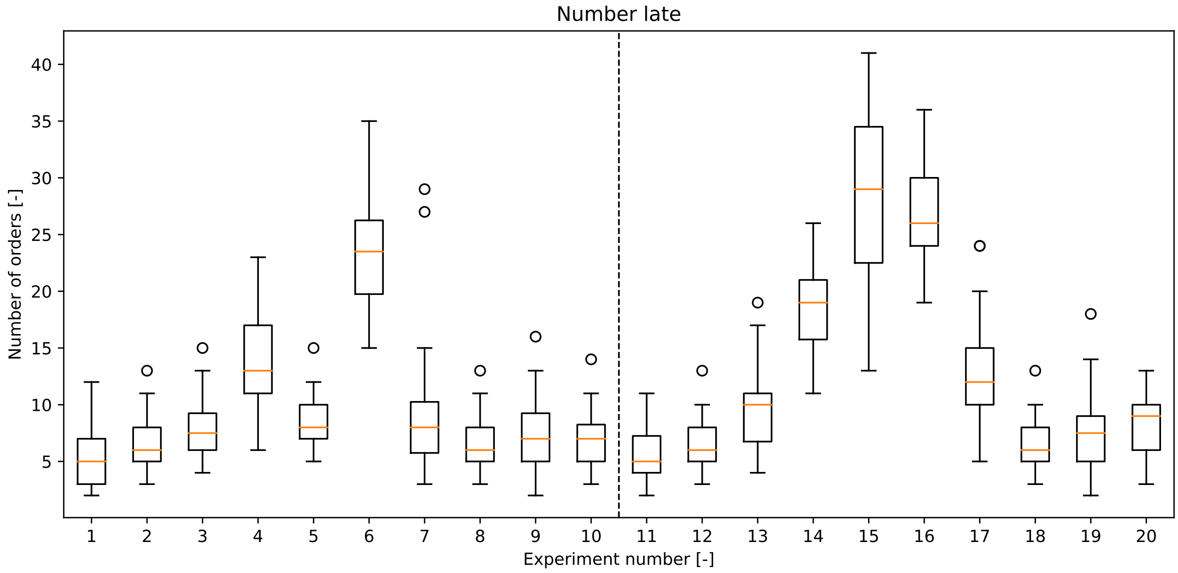

Figure E.17: Results group size: Number late

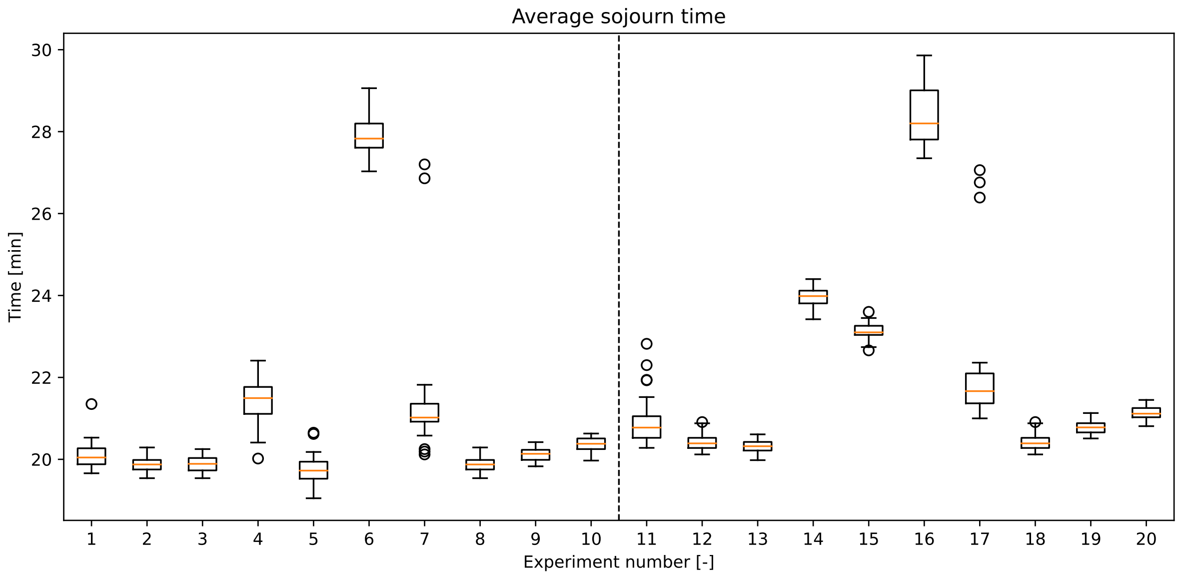

Figure E.18: Results group size: Sojourn time

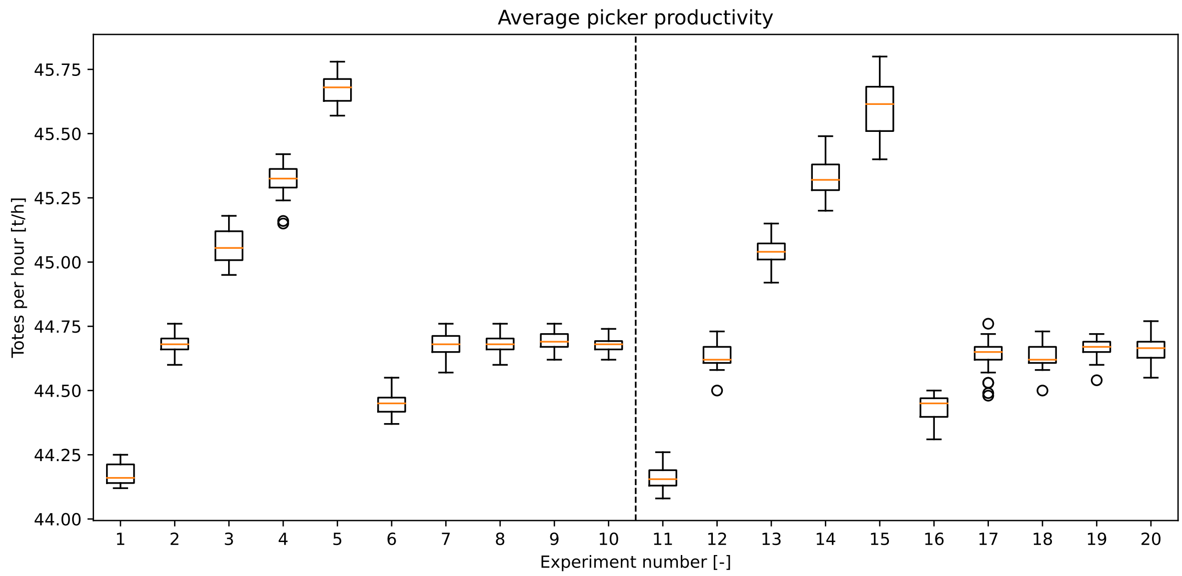

Figure E.19: Results group size: Picker productivity

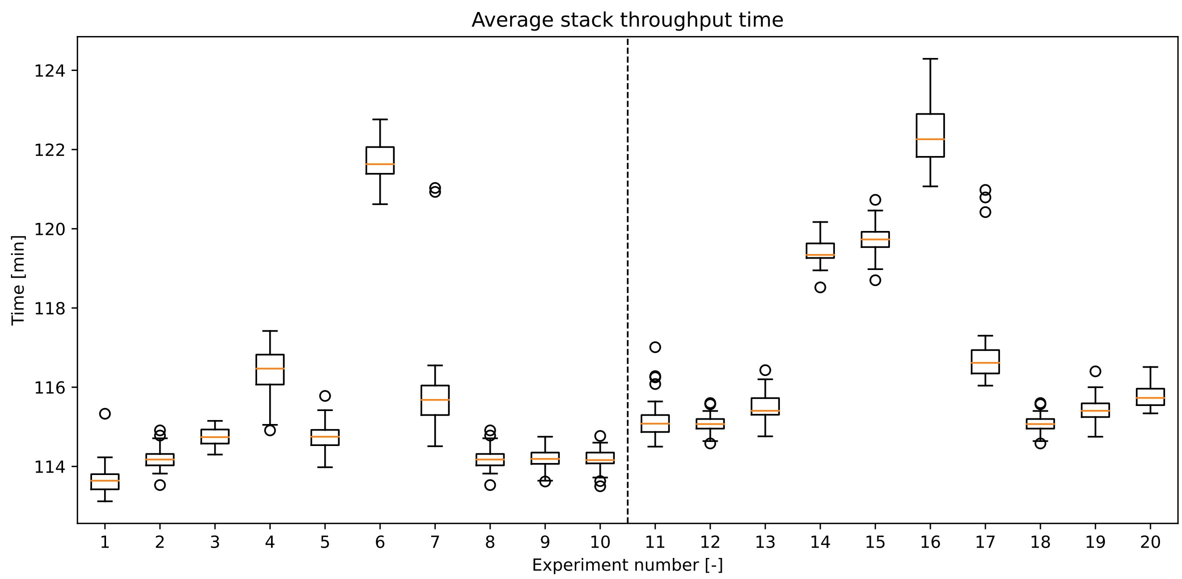

Figure E.20: Results group size: Stack throughput time

The combination between group size and strategy has also been tested to see if they impact each other in any significant way. The experimental design can be found in Table D.6.

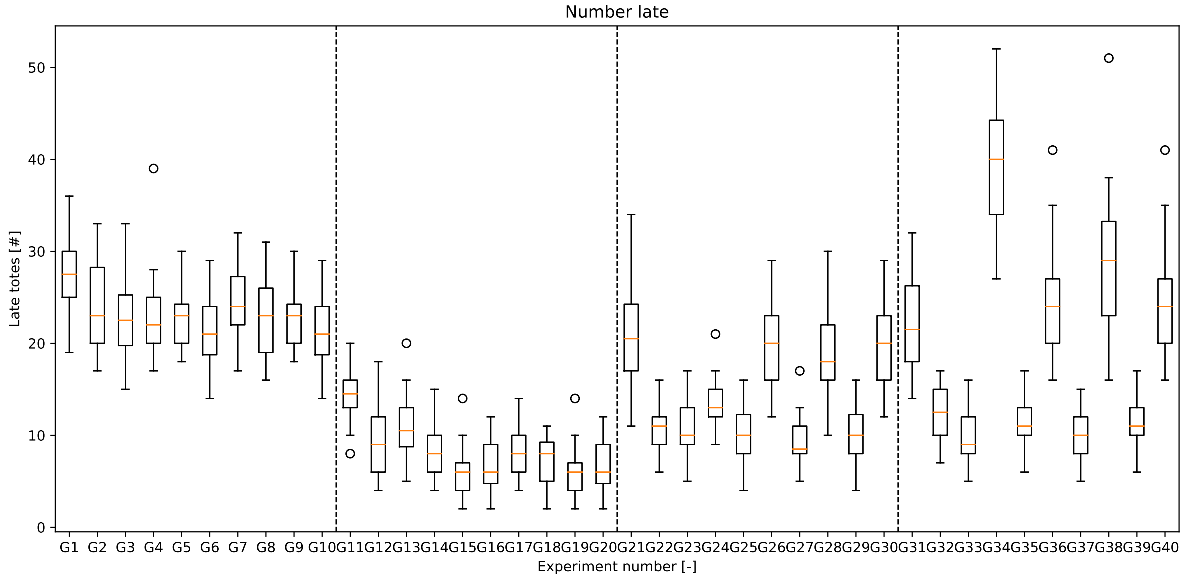

Figure E.21: Results group size and strategy: Number late

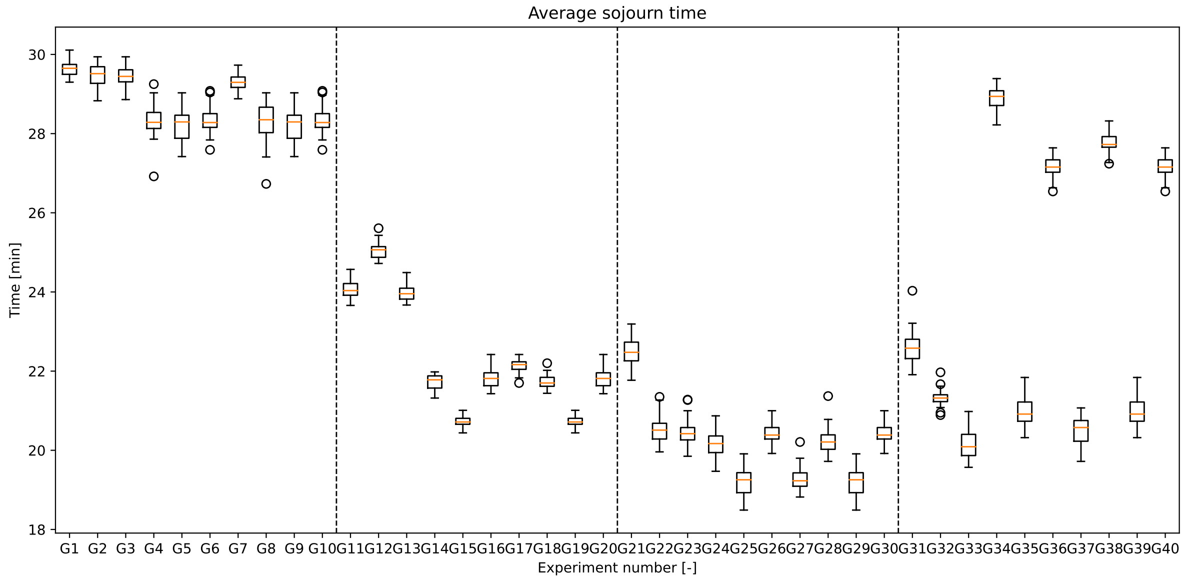

Figure E.22: Results group size and strategy: Sojourn time

What becomes clear from the number of late totes is that small max group size is not preferred. Combined with a small min group size, it results in many late totes for all strategies. With a high min group size, it performs a bit better with some strategies, but it is still inferior to the experiments with a high max group size. When looking at late totes, it is best to have a low minimum group size and a higher maximum group size. When looking at the sojourn time, it may be better to use a higher group size. The picker productivity is about 4% higher when the minimum group size is higher, which is to be expected. To compare these results against a less busy day, these experiments have also been run with the small dataset.

As can be seen from the results with the small dataset, not a lot changes except the strategies that prioritise max group size give worse results. Something that makes sense and on a less busy day it

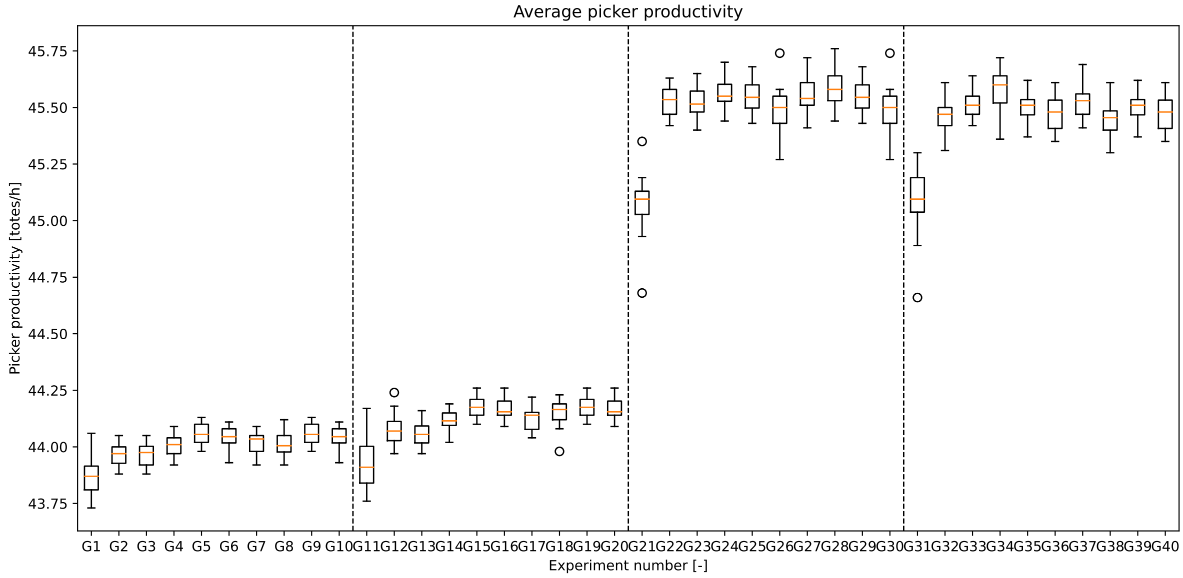

Figure E.23: Results group size and strategy: Picker productivity

will take longer for the buffers to fill up to the max group size and it will take longer for pickers to find a group for which they can start a pick round.

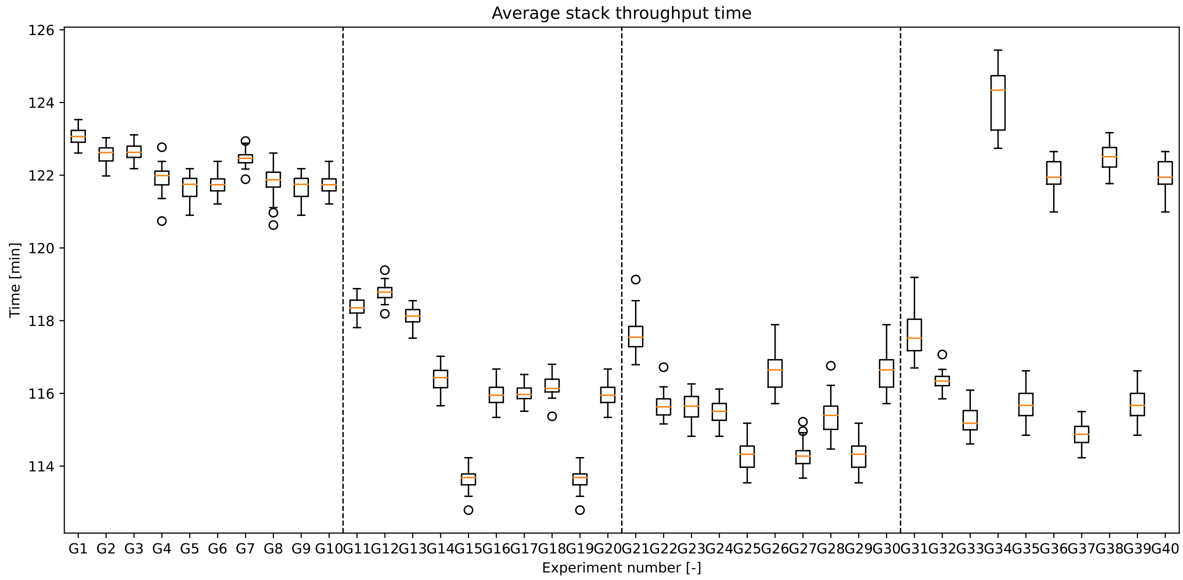

Figure E.24: Results group size and strategy: Stack throughput time

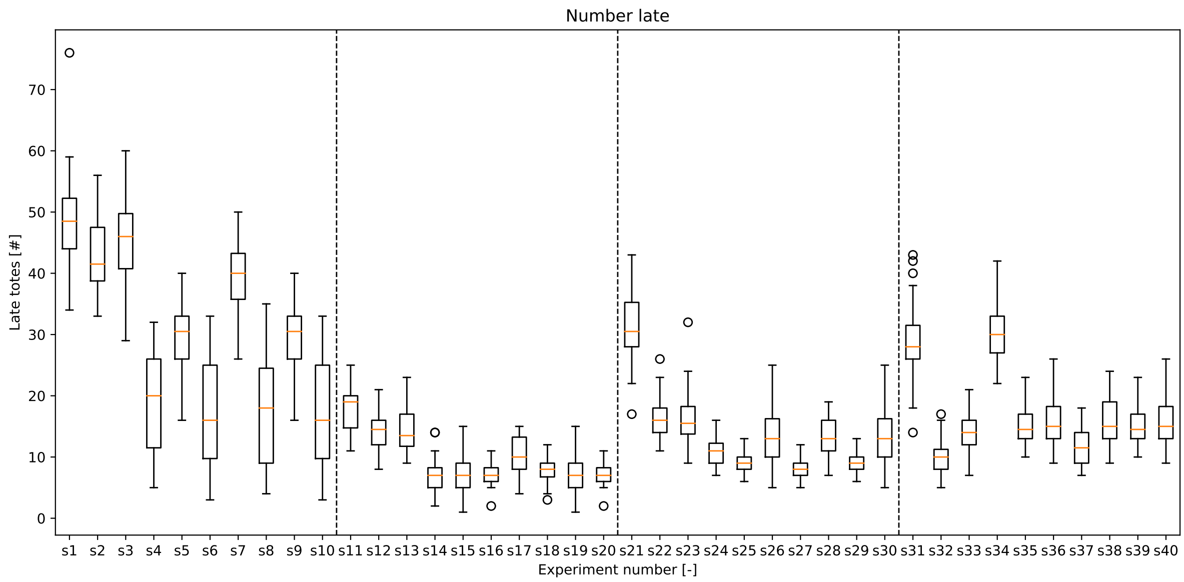

Figure E.25: Results group size and strategy: Number late

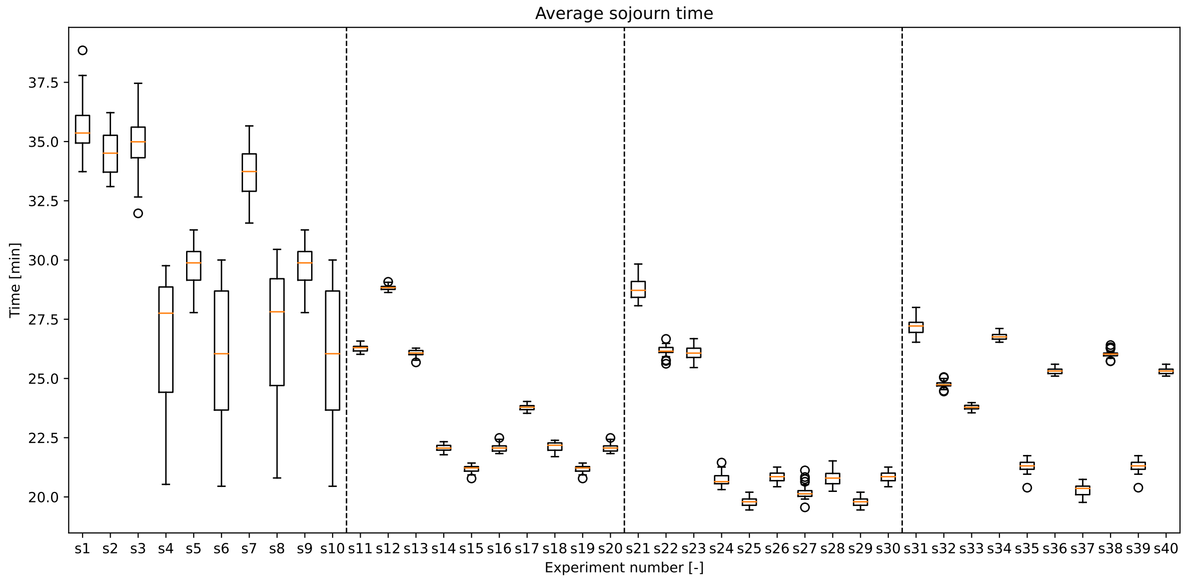

Figure E.26: Results group size and strategy: Sojourn time

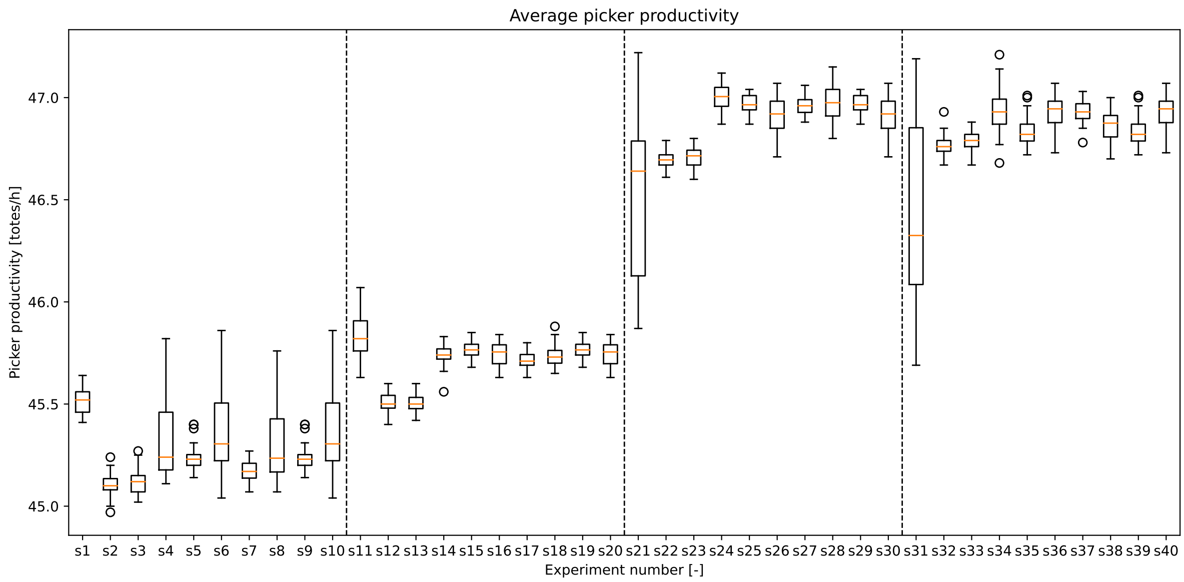

Figure E.27: Results group size and strategy: Picker productivity

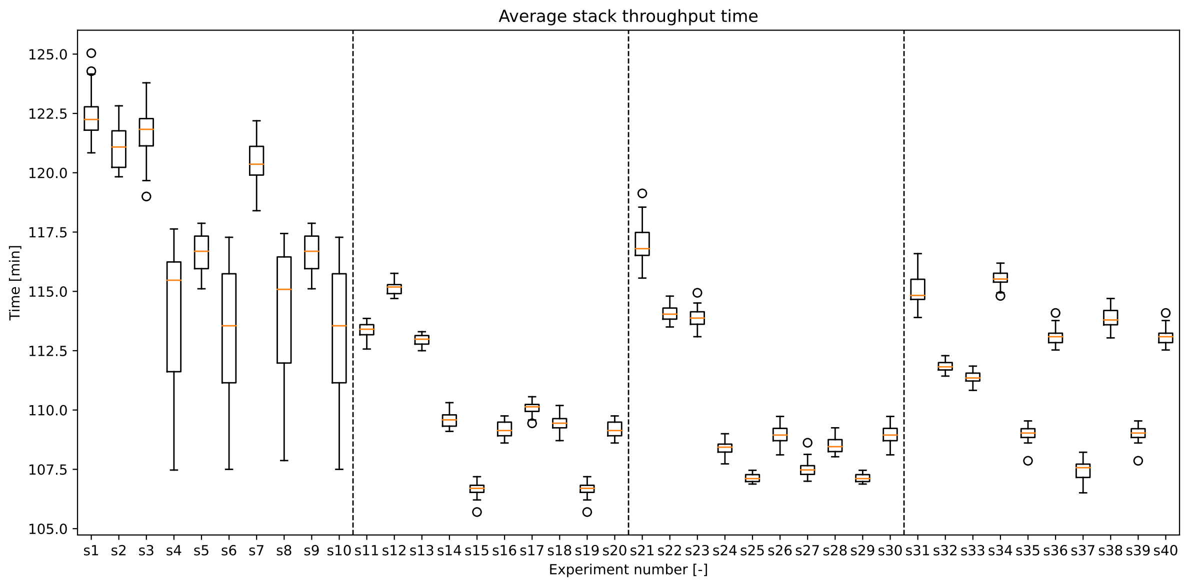

Figure E.28: Results group size and strategy: Stack throughput time

E.5. Consolidation outage

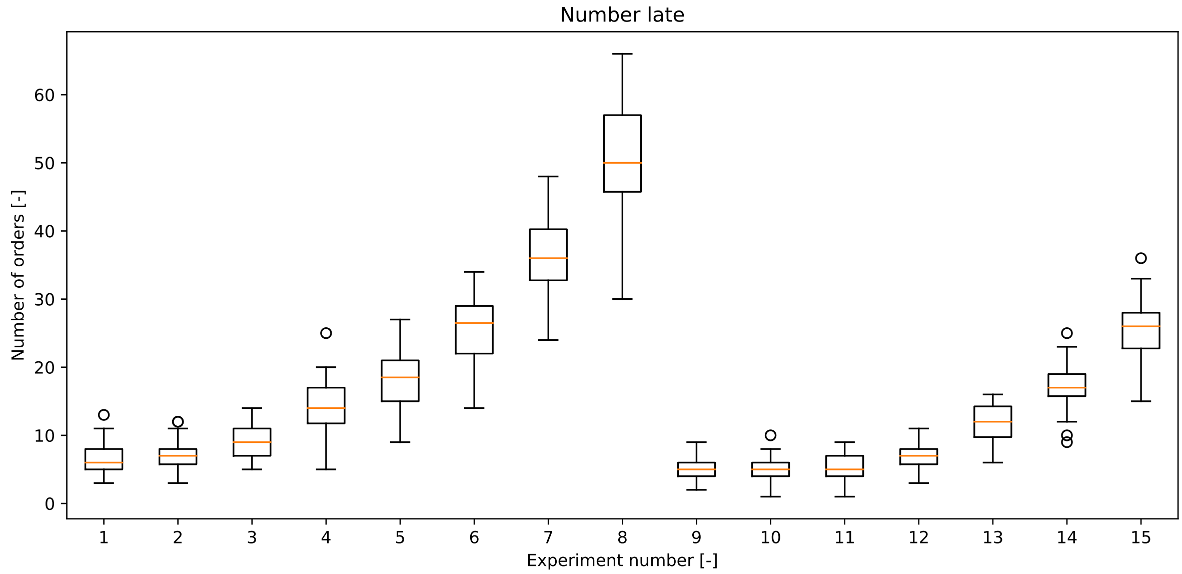

Figure E.29: Results consolidation outage: Number late

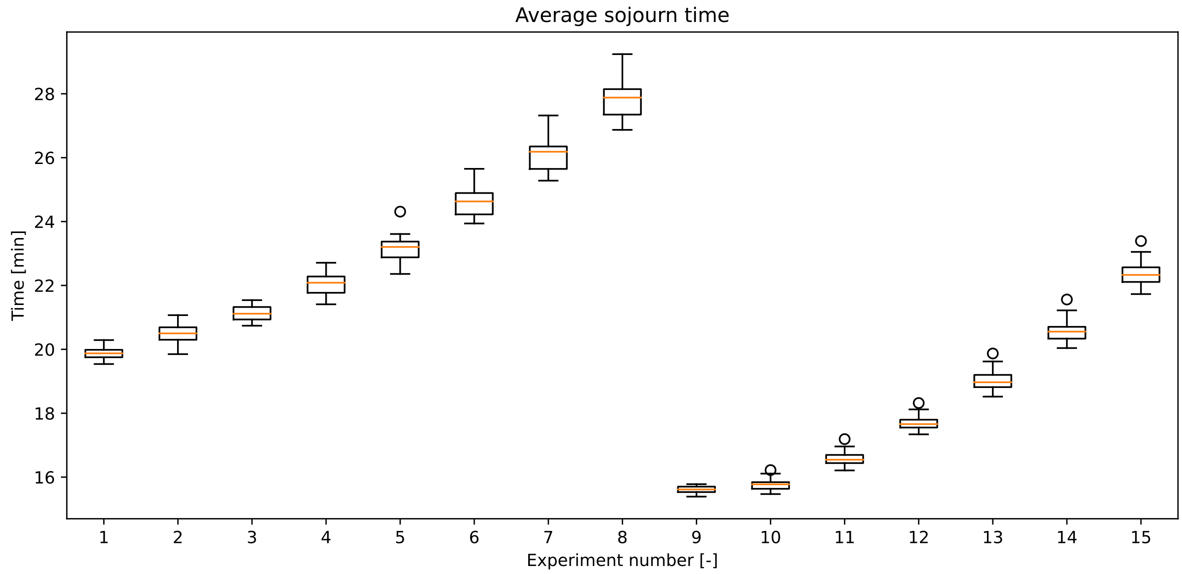

Figure E.30: Results consolidation outage: Sojourn time

Because the facility in Dordrecht has a capacity of 20 totes in the buffer lanes, the output of the maximum buffer items is also shown in Table E.1. This table, in combination with the results of the other KPIs, allows for a reasonable conclusion on the maximum time a consolidation station can be missed. That output is shown in Table E.1.

As can be seen from Figure E.29 the number of late totes gradually increases as the number of minutes that a consolidation station is broken increases. As this is an important KPI, the number of minutes that a consolidation station can be broken should always be minimised. When looking at the values for a situation where there is an abundance of pickers, it can be seen that the abundance of pickers can only partially capture the missing capacity of the broken consolidation station. This is to be expected as pickers can't take over the job of the consolidation station but it does mean that the average group size a picker does a pick round for will be smaller.

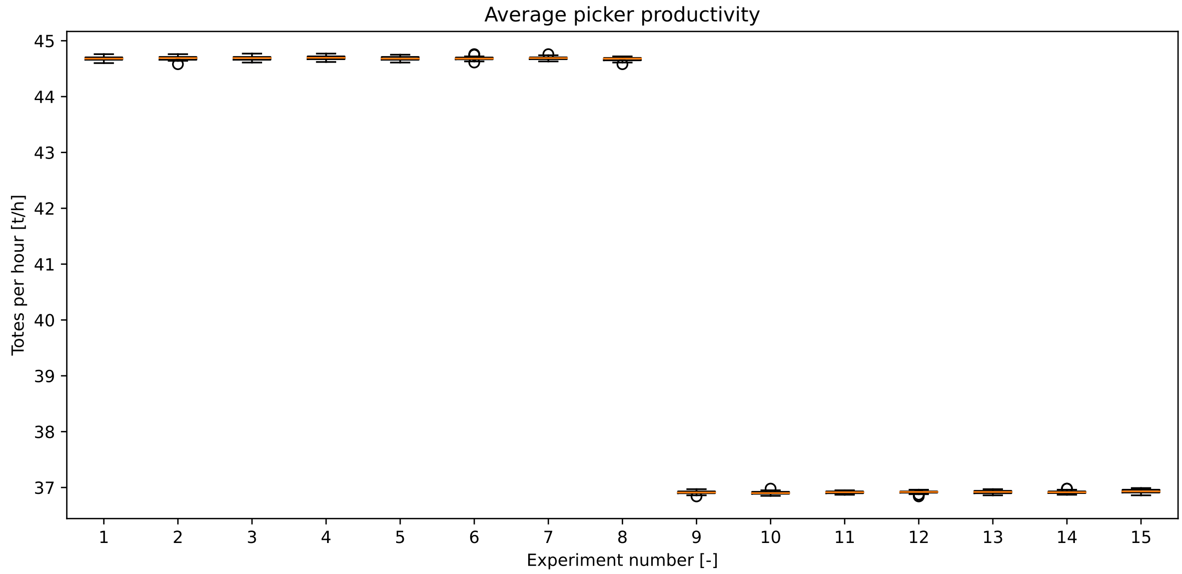

Figure E.31: Results consolidation outage: Picker productivity

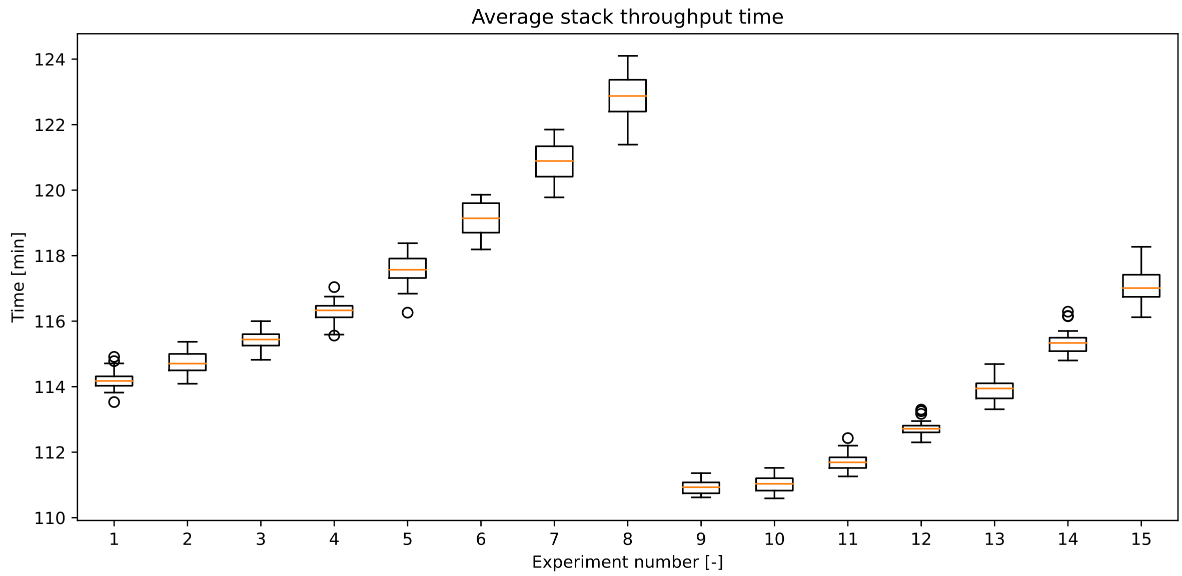

Figure E.32: Results consolidation outage: Stack throughput time

Table E.1: Results consolidation outage: maximum buffer lane items

Ex#	Data item	Statistic	Average	Minimum	Maximum	Half width
1	**buffer items (#)**	**Maximum**	15,8125	15	17	0,169788
2	**buffer items (#)**	**Maximum**	17,34375	16	19	0,268702
3	**buffer items (#)**	**Maximum**	20,375	19	22	0,300255
4	**buffer items (#)**	**Maximum**	23,28125	22	25	0,262785
5	**buffer items (#)**	**Maximum**	25,6875	24	28	0,309707
6	**buffer items (#)**	**Maximum**	26,71875	25	29	0,278285
7	**buffer items (#)**	**Maximum**	29,03125	28	31	0,296522
8	**buffer items (#)**	**Maximum**	31,15625	30	33	0,305232
9	**buffer items (#)**	**Maximum**	15	15	15	-
10	**buffer items (#)**	**Maximum**	15	15	15	-
11	**buffer items (#)**	**Maximum**	17,3125	16	19	0,213543
12	**buffer items (#)**	**Maximum**	20,5	19	22	0,24229
13	**buffer items (#)**	**Maximum**	24	23	25	0,183154
14	**buffer items (#)**	**Maximum**	25,9375	25	28	0,241206
15	**buffer items (#)**	**Maximum**	28,75	27	30	0,259018
16	**buffer items (#)**	**Maximum**	31,75	31	33	0,224317

F

Raw output

F.1. Strategies

Table F.1: Raw results strategies 1.1

Experiment 1	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,14875	21,49	22,55	0,098749
	Maximum	85,49	67,28	100,82	2,512449
IPB time (min)	Average	17,76813	17,13	18,16	0,09333
	Maximum	83,98969	65,55	98,38	2,4873
Number late (#)	Total	19,125	14	28	1,21749
stack throughput time (min)	Average	117,3613	116,64	118,07	0,133287
	Maximum	183,6869	160,48	202,68	3,560881
picker productivity (#/h)	Average	44,34875	43,91	44,8	0,082824
	Minimum	41,85594	40,27	43,12	0,25791
	Maximum	46,61125	45,55	48,2	0,250955
buffer items (#)	Maximum	17,6875	17	19	0,213543

Experiment 2	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	21,97219	21,65	22,17	0,047811
	Maximum	138,3719	92,14	287,08	12,1136
IPB time (min)	Average	17,29938	16,98	17,49	0,047337
	Maximum	131,5847	86,41	280,18	12,16565
Number late (#)	Total	24,40625	11	36	2,222243
stack throughput time (min)	Average	118,345	117,3	119,86	0,201629
	Maximum	196,6594	162,96	334,93	11,88575
picker productivity (#/h)	Average	44,62156	44,56	44,69	0,01166
	Minimum	42,26719	41,02	43,08	0,184029
	Maximum	47,12656	45,67	49,28	0,301888
buffer items (#)	Maximum	17,8125	16	19	0,232351

Experiment 3	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	21,69844	21,1	22,11	0,087398
	Maximum	50,88219	47,73	55,38	0,674315
IPB time (min)	Average	17,34938	16,8	17,73	0,08345
	Maximum	43,64781	40,59	47,58	0,690633
Number late (#)	Total	12,53125	6	20	1,21485
stack throughput time (min)	Average	116,5256	116,01	117	0,104076
	Maximum	160,0072	156,94	166,29	0,903279
picker productivity (#/h)	Average	44,60188	44,52	44,7	0,014111
	Minimum	42,07344	39,84	43,52	0,30722
	Maximum	46,91938	45,64	49,36	0,270914
buffer items (#)	Maximum	17,71875	17	19	0,209516

Experiment 4	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,62125	20,29	20,95	0,053073
	Maximum	83,785	47,09	116,23	6,284475
IPB time (min)	Average	15,8525	15,53	16,16	0,048803
	Maximum	82,4075	41,17	115,54	6,57362
Number late (#)	Total	10,90625	6	16	0,917413
stack throughput time (min)	Average	115,7306	115,27	116,13	0,077153
	Maximum	165,04	156,93	209,58	3,96443
picker productivity (#/h)	Average	44,58094	44,46	44,71	0,018163
	Minimum	41,7175	39,95	43	0,255706
	Maximum	47,19719	45,95	49,02	0,245496
buffer items (#)	Maximum	16,3125	16	17	0,169788

Table F.2: Raw results strategies 1.2

Experiment 5	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,08156	19,59	20,55	0,06655
	Maximum	45,485	38,91	54,61	1,272687
IPB time (min)	Average	15,40375	14,92	15,84	0,065513
	Maximum	38,19625	32,96	49,3	1,153524
Number late (#)	Total	9,25	4	18	1,218351
stack throughput time (min)	Average	114,6988	114,09	115,34	0,103063
	Maximum	159,3566	156,23	166,61	0,844798
picker productivity (#/h)	Average	44,66625	44,61	44,74	0,01214
	Minimum	41,88656	40,42	43,37	0,270362
	Maximum	47,28656	46,18	48,75	0,256581
buffer items (#)	Maximum	16,1875	15	18	0,232351

Experiment 6	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,65906	20,31	21,13	0,083181
	Maximum	50,87125	46,4	71,54	1,590316
IPB time (min)	Average	15,90344	15,57	16,33	0,077184
	Maximum	44,09125	39,63	70,45	1,959329
Number late (#)	Total	10,8125	4	16	1,216413
stack throughput time (min)	Average	115,5797	114,94	116,13	0,116538
	Maximum	162,0494	156,2	187,31	2,199937
picker productivity (#/h)	Average	44,63156	44,56	44,71	0,013709
	Minimum	41,85125	40,23	43,02	0,29321
	Maximum	47,235	46,09	48,11	0,20329
buffer items (#)	Maximum	16,4375	15	18	0,241206

Experiment 7	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	21,3725	20,88	21,76	0,095688
	Maximum	50,97375	45,21	58,11	1,044348
IPB time (min)	Average	16,95188	16,49	17,34	0,091702
	Maximum	43,54375	39,26	49,94	1,022772
Number late (#)	Total	12,0625	7	22	1,238617
stack throughput time (min)	Average	115,8891	115,33	116,61	0,121943
	Maximum	161,4903	157,74	178,7	1,510262
picker productivity (#/h)	Average	44,645	44,59	44,7	0,00999
	Minimum	42,07375	41,09	43,27	0,257277
	Maximum	47,00063	45,83	48,38	0,210021
buffer items (#)	Maximum	17,625	17	19	0,219594

Experiment 8	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,61469	20,09	20,95	0,067482
	Maximum	57,86281	45,75	100,28	5,258143
IPB time (min)	Average	15,85281	15,37	16,16	0,061187
	Maximum	54,41531	38,57	99,91	5,91998
Number late (#)	Total	10,34375	3	16	1,120122
stack throughput time (min)	Average	115,6303	114,82	116,16	0,095224
	Maximum	162,6353	156,21	198,41	3,08399
picker productivity (#/h)	Average	44,58813	44,5	44,66	0,015418
	Minimum	41,68469	39,66	42,69	0,28112
	Maximum	47,32656	45,69	50,11	0,275697
buffer items (#)	Maximum	16,59375	16	18	0,201872

Table F.3: Raw results strategies 1.3

Experiment 9	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,08156	19,59	20,55	0,06655
	Maximum	45,485	38,91	54,61	1,272687
IPB time (min)	Average	15,40375	14,92	15,84	0,065513
	Maximum	38,19625	32,96	49,3	1,153524
Number late (#)	Total	9,25	4	18	1,218351
stack throughput time (min)	Average	114,6988	114,09	115,34	0,103063
	Maximum	159,3566	156,23	166,61	0,844798
picker productivity (#/h)	Average	44,66625	44,61	44,74	0,01214
	Minimum	41,88656	40,42	43,37	0,270362
	Maximum	47,28656	46,18	48,75	0,256581
buffer items (#)	Maximum	16,1875	15	18	0,232351

Experiment 10	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,65906	20,31	21,13	0,083181
	Maximum	50,87125	46,4	71,54	1,590316
IPB time (min)	Average	15,90344	15,57	16,33	0,077184
	Maximum	44,09125	39,63	70,45	1,959329
Number late (#)	Total	10,8125	4	16	1,216413
stack throughput time (min)	Average	115,5797	114,94	116,13	0,116538
	Maximum	162,0494	156,2	187,31	2,199937
picker productivity (#/h)	Average	44,63156	44,56	44,71	0,013709
	Minimum	41,85125	40,23	43,02	0,29321
	Maximum	47,235	46,09	48,11	0,20329
buffer items (#)	Maximum	16,4375	15	18	0,241206

Experiment 11	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,93719	22,39	23,38	0,075684
	Maximum	85,49094	68,72	93,64	1,994986
IPB time (min)	Average	18,44938	17,92	18,89	0,072363
	Maximum	83,8775	67,35	91,11	1,917459
Number late (#)	Total	18,375	9	25	1,574882
stack throughput time (min)	Average	118,0303	117,15	118,93	0,141558
	Maximum	184,0163	159,4	196,35	2,84795
picker productivity (#/h)	Average	44,3625	43,97	44,69	0,077104
	Minimum	41,57813	39,04	43,14	0,349191
	Maximum	46,69656	45,53	48,17	0,239057
buffer items (#)	Maximum	18,21875	17	19	0,219295

Experiment 12	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,84594	22,38	23,68	0,105845
	Maximum	48,97	43,27	54,11	0,855709
IPB time (min)	Average	18,0775	17,58	18,94	0,107835
	Maximum	42,4425	38,47	47,46	0,77832
Number late (#)	Total	9,53125	4	17	1,200965
stack throughput time (min)	Average	117,2281	116,59	118,14	0,128301
	Maximum	160,0397	156,44	167,02	0,787672
picker productivity (#/h)	Average	44,6275	44,54	44,69	0,014451
	Minimum	41,77688	40,24	42,69	0,274934
	Maximum	47,26031	46,34	48,55	0,198379
buffer items (#)	Maximum	18	17	19	0,204772

Table F.4: Raw results strategies 1.4

Experiment 13	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,51125	22,07	23,04	0,07342
	Maximum	48,92406	43,91	53,5	0,759804
IPB time (min)	Average	18,05406	17,62	18,56	0,070144
	Maximum	43,18938	39,42	47,31	0,7915
Number late (#)	Total	9,9375	5	16	0,882838
stack throughput time (min)	Average	117,0931	116,61	117,57	0,094517
	Maximum	159,4284	157,13	167,21	0,808755
picker productivity (#/h)	Average	44,61438	44,54	44,67	0,010677
	Minimum	41,78094	40,61	42,77	0,209075
	Maximum	46,9475	45,69	47,74	0,174218
buffer items (#)	Maximum	18,0625	17	19	0,241206

Experiment 14	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,48406	20,21	20,8	0,056806
	Maximum	76,71906	48,82	106,39	6,585192
IPB time (min)	Average	15,73031	15,46	16,03	0,055307
	Maximum	75,73656	45,25	105,99	6,66621
Number late (#)	Total	9,03125	5	17	1,050081
stack throughput time (min)	Average	115,4263	114,95	115,91	0,092926
	Maximum	165,06	156,52	192,54	3,412951
picker productivity (#/h)	Average	44,5775	44,48	44,71	0,019034
	Minimum	41,48375	39,97	42,81	0,265377
	Maximum	47,23656	46,08	49,1	0,265738
buffer items (#)	Maximum	16,25	15	17	0,183154

Experiment 15	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	19,87844	19,54	20,29	0,064006
	Maximum	41,1725	35,87	48,4	1,286069
IPB time (min)	Average	15,15625	14,83	15,56	0,060868
	Maximum	34,44531	27,18	41,4	1,381891
Number late (#)	Total	6,625	3	13	0,881947
stack throughput time (min)	Average	114,1947	113,53	114,91	0,103283
	Maximum	158,6469	156,32	164,24	0,645276
picker productivity (#/h)	Average	44,68094	44,6	44,76	0,011488
	Minimum	42,01594	40,71	43,33	0,246926
	Maximum	47,23719	46,11	48,8	0,260671
buffer items (#)	Maximum	15,8125	15	17	0,169788

Experiment 16	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,41375	20,12	20,91	0,071323
	Maximum	45,93344	40,88	52,15	1,029219
IPB time (min)	Average	15,65469	15,32	16,12	0,068945
	Maximum	39,53406	34,7	50,55	1,275133
Number late (#)	Total	6,625	3	13	0,791756
stack throughput time (min)	Average	115,0769	114,58	115,6	0,083216
	Maximum	161,4981	155,99	168,88	1,669921
picker productivity (#/h)	Average	44,63281	44,5	44,73	0,016464
	Minimum	41,71844	40,28	43,53	0,262813
	Maximum	47,36844	46,14	48,99	0,223624
buffer items (#)	Maximum	16,21875	15	18	0,219295

Table F.5: Raw results strategies 1.5

Experiment 17	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	21,67438	21,2	22,31	0,088052
	Maximum	47,20313	40,05	54,17	1,272856
IPB time (min)	Average	17,0875	16,61	17,72	0,087037
	Maximum	41,61313	34,79	47,36	1,114313
Number late (#)	Total	8,25	4	16	1,091264
stack throughput time (min)	Average	115,9972	115,3	116,69	0,119914
	Maximum	159,0438	156,94	161,23	0,512054
picker productivity (#/h)	Average	44,66406	44,59	44,72	0,011865
	Minimum	41,91438	39,91	43,34	0,304282
	Maximum	47,05375	46,12	48,27	0,177819
buffer items (#)	Maximum	17,28125	17	18	0,164695

Experiment 18	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,41844	20,12	20,9	0,071595
	Maximum	54,69438	43,45	82,13	3,867807
IPB time (min)	Average	15,66969	15,4	16,13	0,068263
	Maximum	51,27531	36,77	80,25	4,583051
Number late (#)	Total	8,0625	3	17	1,197304
stack throughput time (min)	Average	115,2509	114,88	115,75	0,083783
	Maximum	159,9241	156,16	171,51	1,343213
picker productivity (#/h)	Average	44,61469	44,5	44,74	0,017009
	Minimum	41,73219	39,53	42,94	0,314755
	Maximum	47,33844	46,36	49,14	0,236671
buffer items (#)	Maximum	16,25	15	17	0,204772

Experiment 19	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	19,87844	19,54	20,29	0,064006
	Maximum	41,1725	35,87	48,4	1,286069
IPB time (min)	Average	15,15625	14,83	15,56	0,060868
	Maximum	34,44531	27,18	41,4	1,381891
Number late (#)	Total	6,625	3	13	0,881947
stack throughput time (min)	Average	114,1947	113,53	114,91	0,103283
	Maximum	158,6469	156,32	164,24	0,645276
picker productivity (#/h)	Average	44,68094	44,6	44,76	0,011488
	Minimum	42,01594	40,71	43,33	0,246926
	Maximum	47,23719	46,11	48,8	0,260671
buffer items (#)	Maximum	15,8125	15	17	0,169788

Experiment 20	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,41375	20,12	20,91	0,071323
	Maximum	45,93344	40,88	52,15	1,029219
IPB time (min)	Average	15,65469	15,32	16,12	0,068945
	Maximum	39,53406	34,7	50,55	1,275133
Number late (#)	Total	6,625	3	13	0,791756
stack throughput time (min)	Average	115,0769	114,58	115,6	0,083216
	Maximum	161,4981	155,99	168,88	1,669921
picker productivity (#/h)	Average	44,63281	44,5	44,73	0,016464
	Minimum	41,71844	40,28	43,53	0,262813
	Maximum	47,36844	46,14	48,99	0,223624
buffer items (#)	Maximum	16,21875	15	18	0,219295

Table F.6: Raw results strategies 1.6

Experiment 21	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	24,07594	23,5	24,94	0,102172
	Maximum	73,73719	60,12	95,52	3,618682
IPB time (min)	Average	19,48156	18,9	20,31	0,0983
	Maximum	70,86375	58,33	94,26	3,969039
Number late (#)	Total	15,21875	10	22	1,034998
stack throughput time (min)	Average	116,5372	115,49	117,59	0,173678
	Maximum	178,8234	161,93	194,91	3,049555
picker productivity (#/h)	Average	44,42531	43,96	44,82	0,066995
	Minimum	41,59625	40,06	42,64	0,234467
	Maximum	46,83438	45,69	48,43	0,239758
buffer items (#)	Maximum	19,09375	18	21	0,280162

Experiment 22	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	24,03844	23,67	24,84	0,087795
	Maximum	67,04156	61,19	84,6	1,735075
IPB time (min)	Average	19,28844	18,93	20,1	0,087843
	Maximum	59,49656	54,13	76,97	1,667122
Number late (#)	Total	8,28125	4	13	0,830448
stack throughput time (min)	Average	116,2494	115,38	116,87	0,126515
	Maximum	161,1922	156,72	172,4	1,513523
picker productivity (#/h)	Average	44,61219	44,55	44,69	0,011428
	Minimum	41,76813	40,32	43,1	0,23437
	Maximum	47,39063	46,02	48,89	0,255537
buffer items (#)	Maximum	19,09375	18	20	0,212003

Experiment 23	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	23,79813	23,36	24,25	0,079432
	Maximum	64,70719	59,45	71,73	1,197149
IPB time (min)	Average	19,22031	18,81	19,66	0,076276
	Maximum	57,44813	51,88	63,78	1,208314
Number late (#)	Total	8,875	4	15	0,905408
stack throughput time (min)	Average	116,1066	115,24	116,67	0,113816
	Maximum	160,8663	156,84	176,1	1,470425
picker productivity (#/h)	Average	44,59656	44,54	44,67	0,013003
	Minimum	41,82531	40,02	42,96	0,285865
	Maximum	47,14125	46,43	48,43	0,178558
buffer items (#)	Maximum	19,15625	19	20	0,133003

Experiment 24	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,96906	20,58	21,41	0,065003
	Maximum	79,02625	50,8	106	5,902449
IPB time (min)	Average	16,26469	15,89	16,66	0,063558
	Maximum	77,9825	46,24	105,49	6,101484
Number late (#)	Total	8,21875	5	12	0,807407
stack throughput time (min)	Average	114,8894	114,3	115,53	0,110865
	Maximum	164,3031	156,48	191,88	3,296914
picker productivity (#/h)	Average	44,57875	44,48	44,69	0,018537
	Minimum	41,38031	39,73	42,68	0,240811
	Maximum	47,60063	46,14	49,44	0,251333
buffer items (#)	Maximum	16,96875	16	18	0,214461

Table F.7: Raw results strategies 1.7

Experiment 25	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,37156	21,36	23,29	0,172867
	Maximum	65,97063	56,4	81,56	1,861746
IPB time (min)	Average	17,67969	16,68	18,59	0,172188
	Maximum	59,2225	49,64	76,51	1,877765
Number late (#)	Total	10,1875	6	19	1,181439
stack throughput time (min)	Average	114,3513	113,54	115,36	0,158186
	Maximum	161,2175	156,76	172,03	1,455345
picker productivity (#/h)	Average	44,6825	44,63	44,76	0,011656
	Minimum	41,62938	39,91	42,72	0,253251
	Maximum	47,22594	46,05	49,11	0,235744
buffer items (#)	Maximum	18,21875	16	20	0,313701

Experiment 26	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	21,21625	20,72	21,77	0,102422
	Maximum	56,86813	49,06	66,43	1,892917
IPB time (min)	Average	16,51563	16,03	17,1	0,102414
	Maximum	50,10156	41,1	60,59	1,7273
Number late (#)	Total	8,4375	5	15	1,079432
stack throughput time (min)	Average	114,6625	113,99	115,45	0,136388
	Maximum	160,565	156,82	168,99	1,314403
picker productivity (#/h)	Average	44,62906	44,55	44,7	0,012023
	Minimum	41,79844	40,59	43,24	0,230441
	Maximum	47,06344	46,18	48,75	0,214965
buffer items (#)	Maximum	16,9375	16	18	0,241206

Experiment 27	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	24,46188	23,69	27,83	0,262739
	Maximum	67,11531	62,89	71,8	0,939927
IPB time (min)	Average	19,79844	19,05	23,11	0,25837
	Maximum	60,22813	55,28	64,9	0,879581
Number late (#)	Total	11,1875	5	19	1,00082
stack throughput time (min)	Average	116,2453	115,46	118,43	0,196182
	Maximum	162,7463	157,4	181,62	1,980881
picker productivity (#/h)	Average	44,62406	44,57	44,69	0,009599
	Minimum	41,77125	39,52	43,11	0,333821
	Maximum	47,24375	46,28	48,78	0,236984
buffer items (#)	Maximum	19,84375	18	24	0,389706

Experiment 28	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,95344	20,58	21,29	0,070672
	Maximum	62,63313	50,95	122,81	5,593002
IPB time (min)	Average	16,24688	15,88	16,58	0,069788
	Maximum	58,735	43,62	121,7	6,060053
Number late (#)	Total	8,09375	3	14	1,115432
stack throughput time (min)	Average	114,8534	114,19	115,53	0,105998
	Maximum	164,2856	156,42	218,74	4,457691
picker productivity (#/h)	Average	44,61813	44,51	44,73	0,016444
	Minimum	41,72625	39,7	43,11	0,302564
	Maximum	47,17344	46,09	49,21	0,271084
buffer items (#)	Maximum	16,8125	16	18	0,249746

Table F.8: Raw results strategies 1.8

Experiment 29	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,37156	21,36	23,29	0,172867
	Maximum	65,97063	56,4	81,56	1,861746
IPB time (min)	Average	17,67969	16,68	18,59	0,172188
	Maximum	59,2225	49,64	76,51	1,877765
Number late (#)	Total	10,1875	6	19	1,181439
stack throughput time (min)	Average	114,3513	113,54	115,36	0,158186
	Maximum	161,2175	156,76	172,03	1,455345
picker productivity (#/h)	Average	44,6825	44,63	44,76	0,011656
	Minimum	41,62938	39,91	42,72	0,253251
	Maximum	47,22594	46,05	49,11	0,235744
buffer items (#)	Maximum	18,21875	16	20	0,313701

Experiment 30	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	21,21625	20,72	21,77	0,102422
	Maximum	56,86813	49,06	66,43	1,892917
IPB time (min)	Average	16,51563	16,03	17,1	0,102414
	Maximum	50,10156	41,1	60,59	1,7273
Number late (#)	Total	8,4375	5	15	1,079432
stack throughput time (min)	Average	114,6625	113,99	115,45	0,136388
	Maximum	160,565	156,82	168,99	1,314403
picker productivity (#/h)	Average	44,62906	44,55	44,7	0,012023
	Minimum	41,79844	40,59	43,24	0,230441
	Maximum	47,06344	46,18	48,75	0,214965
buffer items (#)	Maximum	16,9375	16	18	0,241206

Experiment 31	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	24,02344	23,21	24,89	0,138268
	Maximum	80,83031	67,22	103,22	3,211858
IPB time (min)	Average	19,46938	18,7	20,29	0,131278
	Maximum	76,44531	61,98	101,51	3,600769
Number late (#)	Total	12,125	7	17	1,1169
stack throughput time (min)	Average	115,4747	114,7	116,54	0,15116
	Maximum	176,4006	163,92	195,33	3,277998
picker productivity (#/h)	Average	44,38438	43,92	44,77	0,079645
	Minimum	41,47438	38,83	42,75	0,315935
	Maximum	46,95906	45,99	48,05	0,18111
buffer items (#)	Maximum	19,34375	18	21	0,235432

Experiment 32	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	23,20594	22,67	23,87	0,115691
	Maximum	76,84563	67,63	89,52	2,276815
IPB time (min)	Average	18,49813	17,96	19,17	0,115968
	Maximum	70,4975	61,21	82,6	2,087162
Number late (#)	Total	7,90625	3	19	1,08107
stack throughput time (min)	Average	114,5744	113,99	115,28	0,129571
	Maximum	161,1638	156,63	178,51	1,757667
picker productivity (#/h)	Average	44,60969	44,52	44,71	0,013223
	Minimum	41,50156	40,16	42,79	0,284971
	Maximum	47,22969	46,19	48,62	0,232988
buffer items (#)	Maximum	18,5625	17	20	0,302862

Table F.9: Raw results strategies 1.9

Experiment 33	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	23,64313	23,2	24,28	0,093965
	Maximum	81,47	69,62	101,6	3,15027
IPB time (min)	Average	19,11844	18,7	19,71	0,089741
	Maximum	74,87281	63,45	94,76	3,052797
Number late (#)	Total	7,9375	4	15	0,942563
stack throughput time (min)	Average	115,0603	114,51	115,69	0,108438
	Maximum	160,1259	156,75	169,89	1,214708
picker productivity (#/h)	Average	44,58688	44,51	44,65	0,010216
	Minimum	42,15594	40,45	43,21	0,249452
	Maximum	46,83313	45,58	48,27	0,202638
buffer items (#)	Maximum	19,3125	18	20	0,213543

Experiment 34	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,58031	20,22	20,9	0,058955
	Maximum	79,68563	50,54	114	6,554133
IPB time (min)	Average	15,88094	15,49	16,19	0,057536
	Maximum	77,95719	43,76	113,1	7,04319
Number late (#)	Total	8,3125	4	15	0,875685
stack throughput time (min)	Average	114,4369	113,94	114,99	0,084447
	Maximum	162,5638	156,64	187,85	2,887715
picker productivity (#/h)	Average	44,58625	44,43	44,71	0,020411
	Minimum	41,52125	40,07	42,74	0,218583
	Maximum	47,27219	46,53	48,56	0,196843
buffer items (#)	Maximum	16,46875	16	17	0,182796

Experiment 35	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	23,82813	22,59	26,38	0,335643
	Maximum	89,22969	78,68	104,47	2,170198
IPB time (min)	Average	19,17219	17,93	21,69	0,333777
	Maximum	82,73063	70,53	99,23	2,227143
Number late (#)	Total	8,9375	5	17	0,994516
stack throughput time (min)	Average	114,2484	112,94	116,14	0,278758
	Maximum	164,4138	156,44	179,91	2,571796
picker productivity (#/h)	Average	44,66531	44,59	44,75	0,013521
	Minimum	41,81156	40,38	43,16	0,262128
	Maximum	47,16063	45,75	49,21	0,241733
buffer items (#)	Maximum	20,0625	18	23	0,564053

Experiment 36	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,81625	20,45	21,22	0,070507
	Maximum	63,33344	53,44	76,71	2,253994
IPB time (min)	Average	16,11906	15,76	16,51	0,070908
	Maximum	57,43313	48,76	76,28	2,27306
Number late (#)	Total	7,1875	3	11	0,729388
stack throughput time (min)	Average	113,9538	113,42	114,51	0,087088
	Maximum	160,5638	156,29	195,47	2,733112
picker productivity (#/h)	Average	44,63	44,56	44,71	0,013737
	Minimum	41,84844	40,78	43,01	0,217171
	Maximum	47,15219	46,16	48,39	0,211066
buffer items (#)	Maximum	16,5625	16	18	0,241206

Table F.10: Raw results strategies 1.10

Experiment 37	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	24,62188	23,98	26,16	0,180133
	Maximum	85,39344	73,27	101,4	2,604926
IPB time (min)	Average	20,01344	19,35	21,52	0,17779
	Maximum	79,19156	67,2	93,88	2,534865
Number late (#)	Total	10,46875	6	17	1,031955
stack throughput time (min)	Average	115,0063	114,26	116,28	0,178755
	Maximum	164,2706	157,08	179,55	2,293858
picker productivity (#/h)	Average	44,6175	44,55	44,69	0,012218
	Minimum	41,93594	40,13	43,16	0,272607
	Maximum	47,09344	46	48,57	0,218387
buffer items (#)	Maximum	20,09375	18	22	0,321947

Experiment 38	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,655	20,33	20,94	0,050765
	Maximum	62,56688	49,97	82,37	3,557922
IPB time (min)	Average	15,94875	15,62	16,25	0,050631
	Maximum	59,2825	43,83	81,71	3,972211
Number late (#)	Total	7,71875	2	14	1,080585
stack throughput time (min)	Average	114,3266	113,81	114,73	0,072777
	Maximum	161,6831	156,56	191,56	3,005396
picker productivity (#/h)	Average	44,61656	44,53	44,72	0,016676
	Minimum	41,35625	39,91	42,86	0,280551
	Maximum	47,21906	46,14	48,86	0,224971
buffer items (#)	Maximum	16,625	15	17	0,199587

Experiment 39	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	23,82813	22,59	26,38	0,335643
	Maximum	89,22969	78,68	104,47	2,170198
IPB time (min)	Average	19,17219	17,93	21,69	0,333777
	Maximum	82,73063	70,53	99,23	2,227143
Number late (#)	Total	8,9375	5	17	0,994516
stack throughput time (min)	Average	114,2484	112,94	116,14	0,278758
	Maximum	164,4138	156,44	179,91	2,571796
picker productivity (#/h)	Average	44,66531	44,59	44,75	0,013521
	Minimum	41,81156	40,38	43,16	0,262128
	Maximum	47,16063	45,75	49,21	0,241733
buffer items (#)	Maximum	20,0625	18	23	0,564053

Experiment 40	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	20,81625	20,45	21,22	0,070507
	Maximum	63,33344	53,44	76,71	2,253994
IPB time (min)	Average	16,11906	15,76	16,51	0,070908
	Maximum	57,43313	48,76	76,28	2,27306
Number late (#)	Total	7,1875	3	11	0,729388
stack throughput time (min)	Average	113,9538	113,42	114,51	0,087088
	Maximum	160,5638	156,29	195,47	2,733112
picker productivity (#/h)	Average	44,63	44,56	44,71	0,013737
	Minimum	41,84844	40,78	43,01	0,217171
	Maximum	47,15219	46,16	48,39	0,211066
buffer items (#)	Maximum	16,5625	16	18	0,241206

F.2. Number of pickers

Table F.11: Raw output: Number of pickers 1/2

Experiment 11	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	22,46063	21,14	24,03	0,236359
	Maximum	49,47719	43,18	56,1	1,098574
IPB time (min)	Average	17,32781	16,02	18,93	0,236929
	Maximum	42,54875	35,74	48,94	1,14779
Number late (#)	Total	11,4375	6	19	1,183212
stack throughput time (min)	Average	116,6394	115,26	118,77	0,288483
	Maximum	159,1475	156,44	162,21	0,572381
picker productivity (#/h)	Average	47,20313	47,13	47,26	0,011779
	Minimum	44,57313	43,1	45,77	0,238667
	Maximum	49,56	48,56	50,39	0,169001
buffer items (#)	Maximum	17,71875	16	20	0,333148

Experiment 12	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	19,87844	19,54	20,29	0,064006
	Maximum	41,1725	35,87	48,4	1,286069
IPB time (min)	Average	15,15625	14,83	15,56	0,060868
	Maximum	34,44531	27,18	41,4	1,381891
Number late (#)	Total	6,625	3	13	0,881947
stack throughput time (min)	Average	114,1947	113,53	114,91	0,103283
	Maximum	158,6469	156,32	164,24	0,645276
picker productivity (#/h)	Average	44,68094	44,6	44,76	0,011488
	Minimum	42,01594	40,71	43,33	0,246926
	Maximum	47,23719	46,11	48,8	0,260671
buffer items (#)	Maximum	15,8125	15	17	0,169788

Experiment 13	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	18,52781	18,24	18,9	0,060154
	Maximum	37,10219	33,77	39,78	0,503141
IPB time (min)	Average	14,17188	13,89	14,51	0,05647
	Maximum	29,17313	25,38	34,07	0,673929
Number late (#)	Total	6,15625	3	12	0,84484
stack throughput time (min)	Average	113,0338	112,55	113,59	0,103111
	Maximum	158,7172	156,59	162	0,542472
picker productivity (#/h)	Average	42,4375	42,35	42,51	0,011065
	Minimum	39,71938	37,33	40,62	0,303897
	Maximum	45,08	44,27	46,41	0,194702
buffer items (#)	Maximum	15,09375	15	16	0,106772

Experiment 14	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	17,34938	17,05	17,66	0,052789
	Maximum	36,51563	33,25	40,66	0,757492
IPB time (min)	Average	13,32406	13,05	13,62	0,049712
	Maximum	28,30156	24,6	33,31	0,91452
Number late (#)	Total	5,90625	3	10	0,751597
stack throughput time (min)	Average	112,2106	111,64	112,63	0,08337
	Maximum	159,4119	157,19	163,63	0,570817
picker productivity (#/h)	Average	40,40813	40,33	40,46	0,010338
	Minimum	37,81656	36,26	38,9	0,244982
	Maximum	43,08094	41,72	45,1	0,277753
buffer items (#)	Maximum	15	15	15	

Table F.12: Raw output: Number of pickers 2/2

Experiment 15	Statistic	Average	Minimum	Maximum	Half width
Sojourn Time (min)	Average	16,45219	16,17	16,73	0,053881
	Maximum	36,94063	31,96	45,74	1,120824
IPB time (min)	Average	12,71156	12,46	12,99	0,049917
	Maximum	28,40875	24,73	37	1,17056
Number late (#)	Total	5,65625	3	11	0,796294
stack throughput time (min)	Average	111,4997	110,93	112,15	0,107104
	Maximum	158,9634	156,17	165,66	0,77277
picker productivity (#/h)	Average	38,58469	38,51	38,67	0,01194
	Minimum	36,02875	33,78	37,42	0,277715
	Maximum	41,19594	40,25	42,67	0,221042
buffer items (#)	Maximum	15	15	15	