Developing a decision support tool for the operation of parallel AS/RS during partial downtime

A case study at Jumbo Supermarkets

ME54035: ME-MME MSc Thesis

L.M. van den Brink

Delft University of Technology

TU Delft

Developing a decision support tool for the operation of parallel AS/RS during partial downtime

A case study at Jumbo Supermarkets

by

L.M. van den Brink

Master Thesis

in partial fulfilment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at the Department Maritime and Transport Technology of Faculty Mechanical, Maritime and Materials Engineering of Delft University of Technology

to be defended publicly on Friday, May 31, 2024, at 10:00 AM

Student number:	4855752	
MSc Track:	Multi-Machine Engineering	
Report Number:	2024.MME.8913	
Thesis committee:	Dr. F. Schulte,	TU Delft, Daily Supervisor
	Ir. M. Duinkerken,	TU Delft, Committee Chair
	Ir. D. Jansen,	Company Supervisor
Date:	16-05-2024	

Preface

This thesis concludes my master's in Mechanical Engineering with the track Multi-Machine Engineering at the Delft University of Technology. For the past six months, I have spent countless hours researching, programming, analysing and writing for this research about downtime in Automated Storage and Retrieval systems. Ever since taking the course in System Analysis and Simulation, I have been interested in integrating the concepts taught there into my master's thesis. This passion, coupled with my interest in logistical systems, which stemmed from my employment at Jumbo Supermarkets, has shaped the focus of this research, which I present with satisfaction, and from which I have learned a lot.

This thesis has reached its current state thanks to the invaluable support of several individuals. First, I would like to thank the teachers of the System Analysis and Simulation course, who also happen to be my academic supervisors for this thesis. My gratitude goes out to Dr. Frederik Schulte for being my daily supervisor, and to Ir. Mark Duinkerken for chairing the graduation committee, both providing me with valuable guidance and feedback. Next to that, I would like to thank Ir. Dik Jansen for being my company supervisor and providing me with this highly interesting case study. I thoroughly enjoyed our weekly meetings and discussions on-topic, and conversations off-topic. Your knowledge, insight and critical feedback helped me and this thesis enormously.

L.M. van den Brink
Delft, May 2024

Abstract

This thesis explores the optimisation of Automated Storage and Retrieval Systems (AS/RS) in modern warehousing, focusing on the minimisation of performance losses during partial downtime in various scenarios. The research is conducted in collaboration with Jumbo Supermarkets, utilising their highly automated distribution centre containing an Order Consolidation Buffer with 4 dual-crane AS/RS in parallel as a case study. In this case study, the effects of partial downtime are investigated and operational policies that can effectively mitigate these impacts are identified.

With the increasing reliance on automation within logistics to enhance efficiency, reduce errors, and lower operational costs, AS/RS have become a crucial component of warehouse operations. However, despite their significant advantages, these systems face challenges, notably in dealing with partial downtimes which can arise from unexpected breakdowns or scheduled maintenance. This research aims to fill the gap in existing literature by providing an exploratory analysis of partial downtime in AS/RS, which has been notably absent from prior studies.

Through the development of a reusable Discrete Event Simulation model in Python, this thesis develops a framework that not only addresses the immediate concerns of Jumbo Supermarkets, but also contributes to the broader field of AS/RS optimisation. The study identifies several operational policies that could potentially minimise the negative effects of partial downtime by minimising delays, upstream system interference and added manual work, and by enhancing robustness and resilience. These policies are developed for two scenarios. In the first scenario, one of the two cranes within an AS/RS is down. The developed policies adjust the workload distribution, determining the portion of workload that remains for this AS/RS and the portion that needs to be redistributed across the other AS/RS. In the second scenario, both cranes within an AS/RS are down. The developed policies offer alternative approaches regarding the handling of pallets that were already in production and initially designated for this AS/RS, but can no longer go there. Some policies involve directly unloading these pallets, while others redistribute them across the other AS/RS. Eventually, results for both scenarios are compared, aiding in the decision of whether one should continue operation with one crane, or stop both cranes to speed up repairs.

Key findings suggest that strategic adjustments to workload distribution among AS/RS during partial downtimes, along with the implementation of specific operational policies, can significantly mitigate performance degradation in certain scenarios. In the experiment where one of the cranes within an AS/RS is down, it is found that during average weeks, the system at Jumbo has sufficient overcapacity to handle the original workload, however, it would be more robust and resilient to reduce the capacity of this AS/RS and diverge some of the workload to the other AS/RS. In general, the higher the overall workload on a day, the more important it becomes to shift the workload away from the AS/RS with just one functioning crane to the other AS/RS. In the experiment where both cranes within an AS/RS are down, it is found that the system at Jumbo has sufficient overcapacity to process the pallets in the other AS/RS instead of having to directly unload them to lower the workload. This reduces the amount of added manual work. When comparing both experiments, it is more beneficial to continue the operation of a single crane instead of stopping both cranes, except for very short downtime. Even though moving the broken crane to its repair position causes a longer overall downtime, delays are significantly reduced since a large part of goods stuck in that AS/RS can then still be retrieved.

All in all, this research provides valuable insights into the dynamics of parallel AS/RS under partial downtime conditions and offers practical guidelines for effective operations during partial downtime.

List of Abbreviations

ABM	Agent-Based Modelling
AIO	All-In-One picking system
AS/RS	Automated Storage and Retrieval System
AV	Autonomous Vehicle
AVS/RS	Autonomous Vehicle Storage and Retrieval System
BOM	Bill of Material
CDC	Central Distribution Centre
CPS	Car Picking System
DC	Distribution Centre
DES	Discrete Event Simulation
DF	Depth-First
DOE	Design of Experiment
EFC	E-Fulfillment Centre
I/O	Input / Output
JIT	Just-In-Time
KPI	Key Performance Indicator
MILP	Mixed Integer Linear Programming
ML	Machine Learning
NDC	National Distribution Centre
NN	Nearest-Neighbour
OCB	Order Consolidation Buffer
OPM	Order Picking Machinery
RC	Rollcage
SBS/RS	Shuttle-based Storage and Retrieval System
SKU	Stock-Keeping Unit
SOQN	Semi-Open Queuing Network

Table 1: List of Abbreviations

List of Figures

1.1 Supply Chain locations of Jumbo Supermarkets (Jumbo Supermarkets, 2022) 2
1.2 DEGREE methodology for simulation projects (Rossetti, 2015) 5

2.1 Graph of AS/RS model results (Witt, 1974) 7
2.2 Keywords classification matrix (Bertolini et al., 2023) 8

3.1 Illustration of parallel AS/RS with specified characteristics 16

5.1 Model components and interaction 25
5.2 Probability density for selection of an order to produce from the production pool 27
5.3 Crane speed over time when maximum speed is or is not reached 28
5.4 Time in seconds difference orders start to be processed compared to the scheduled start time 29

6.1 Overview of pallets and trips processed for each day in the week 34
6.2 Spread of departure times throughout the day for each dataset 35
6.3 Spread of release times of trips throughout the day for each dataset 35
6.4 Representation of order lead times 35
6.5 Total warehouse system (Azadeh et al., 2019) 36
6.6 Production time distribution 37
6.7 Buffer side view 37
6.8 Buffer top view 38
6.9 Loading time per pallet and loading time factor 38
6.10 Rack Configuration 40
6.11 Visual representation of locations where containers can be temporarily stored 41

7.1 Number of pallets at different stages of the system throughout the simulation 43
7.2 Realised samples from distributions used 45
7.3 Cranes behavior 46
7.4 Rack behavior 46
7.5 General behaviour 47
7.6 Fill grades of analytical model compared to simulation model 49
7.7 Average dual-command crane cycle 50

8.1 Warm-up period of simulation on Friday 51
8.2 Number of pallets on conveyor during and after downtime in the worst-case scenario . . 52
8.3 KPIs of cranes and rack after downtime on peak day 53

9.1 Remaining crane utilisation experiment 1, day 2, downtime 13:00 - 22:15 59
9.2 Remaining crane utilisation experiment 1 day 3 59
9.3 System fill grade experiment 1 day 3 60
9.4 Pallets on conveyor over time Experiment 2, Day 3, Downtime 13:00 - 21:00, Policy 0, no action 62
9.5 Number of pallets on conveyor, experiment 1, day 3, downtime 07:00 - 15:00, policy 2, store in partner 63
9.6 Crane utilisation experiment 2, day 3, downtime 13:00 - 21:00, policy 2, store in partner 64

E.1 Input data format 135

List of Tables

1 List of Abbreviations . . . v

2.1 Papers studying topics related to this research . . . 14

4.1 Policies when one of the two cranes within an AS/RS is down . . . 17
4.2 Policies when both cranes down . . . 19
4.3 Policy performance evaluation criteria . . . 20

5.1 Input data for each order . . . 24
5.2 Pallet attributes in model . . . 25
5.3 Rack attributes in model . . . 25

6.1 Main simplifications of model . . . 39

8.1 Allowed errors for each KPI . . . 54

9.1 Averaged scores experiment 1 day 1 . . . 58
9.2 Averaged scores experiment 1 day 2 . . . 58
9.3 Averaged scores experiment 1 day 3 . . . 60
9.4 Averaged scores experiment 1 across all days and scenarios . . . 61
9.5 Averaged scores experiment 2 day 1 . . . 62
9.6 Averaged scores experiment 2 day 2 . . . 63
9.7 Averaged scores experiment 2 day 3 . . . 64
9.8 Averaged scores experiment 2 across all days and scenarios . . . 64
9.9 Performance metric comparison of best-performing policies for both experiments . . . 65
9.10 Score comparison of best-performing policies for both experiments . . . 65
9.11 Performance metric comparison of best-performing policies for both experiments with 1h of repair time . . . 66
9.12 Ranking of policies with all possible weight combinations with an allowed weight deviation of 1 . . . 67

C.1 Simulation settings . . . 109
C.2 Crane settings . . . 110
C.3 Rack configuration settings . . . 110
C.4 General rack settings . . . 111

D.1 Results without downtime . . . 113
D.2 Results Experiment 1 Day 1 Downtime 05:00 - 09:15 . . . 114
D.3 Results Experiment 1 Day 1 Downtime 05:00 - 14:15 . . . 114
D.4 Results Experiment 1 Day 1 Downtime 07:00 - 11:15 . . . 114
D.5 Results Experiment 1 Day 1 Downtime 07:00 - 16:15 . . . 115
D.6 Results Experiment 1 Day 1 Downtime 13:00 - 17:15 . . . 115
D.7 Results Experiment 1 Day 1 Downtime 13:00 - 22:15 . . . 115
D.8 Averaged results Experiment 1 Day 1 . . . 116
D.9 Individual scores Experiment 1 Day 1 . . . 116
D.10 Results Experiment 1 Day 2 Downtime 05:00 - 09:15 . . . 117
D.11 Results Experiment 1 Day 2 Downtime 05:00 - 14:15 . . . 117
D.12 Results Experiment 1 Day 2 Downtime 07:00 - 11:15 . . . 117
D.13 Results Experiment 1 Day 2 Downtime 07:00 - 16:15 . . . 118
D.14 Results Experiment 1 Day 2 Downtime 13:00 - 17:15 . . . 118

D.15 Results Experiment 1 Day 2 Downtime 13:00 - 22:15 118
D.16 Averaged results Experiment 1 Day 2 119
D.17 Individual scores Experiment 1 Day 2 119
D.18 Results Experiment 1 Day 3 Downtime 05:00 - 09:15 120
D.19 Results Experiment 1 Day 3 Downtime 05:00 - 14:15 120
D.20 Results Experiment 1 Day 3 Downtime 07:00 - 11:15 120
D.21 Results Experiment 1 Day 3 Downtime 07:00 - 16:15 121
D.22 Results Experiment 1 Day 3 Downtime 13:00 - 17:15 121
D.23 Results Experiment 1 Day 3 Downtime 13:00 - 22:15 121
D.24 Averaged results Experiment 1 Day 3 122
D.25 Individual scores Experiment 1 Day 3 122
D.26 Results Experiment 2 Day 1 Downtime 05:00 - 08:00 123
D.27 Results Experiment 2 Day 1 Downtime 05:00 - 13:00 123
D.28 Results Experiment 2 Day 1 Downtime 07:00 - 10:00 123
D.29 Results Experiment 2 Day 1 Downtime 07:00 - 15:00 124
D.30 Results Experiment 2 Day 1 Downtime 13:00 - 16:00 124
D.31 Results Experiment 2 Day 1 Downtime 13:00 - 21:00 124
D.32 Averaged results Experiment 2 Day 1 125
D.33 Individual scores Experiment 2 Day 1 125
D.34 Results Experiment 2 Day 2 Downtime 05:00 - 08:00 126
D.35 Results Experiment 2 Day 2 Downtime 05:00 - 13:00 126
D.36 Results Experiment 2 Day 2 Downtime 07:00 - 10:00 126
D.37 Results Experiment 2 Day 2 Downtime 07:00 - 15:00 127
D.38 Results Experiment 2 Day 2 Downtime 13:00 - 16:00 127
D.39 Results Experiment 2 Day 2 Downtime 13:00 - 21:00 127
D.40 Averaged results Experiment 2 Day 2 128
D.41 Individual scores Experiment 2 Day 2 128
D.42 Results Experiment 2 Day 3 Downtime 05:00 - 08:00 129
D.43 Results Experiment 2 Day 3 Downtime 05:00 - 13:00 129
D.44 Results Experiment 2 Day 3 Downtime 07:00 - 10:00 129
D.45 Results Experiment 2 Day 3 Downtime 07:00 - 15:00 130
D.46 Results Experiment 2 Day 3 Downtime 13:00 - 16:00 130
D.47 Results Experiment 2 Day 3 Downtime 13:00 - 21:00 130
D.48 Averaged results Experiment 2 Day 3 131
D.49 Individual scores Experiment 2 Day 3 131
D.50 Results with 1 hour of repair time Experiment 1 132
D.51 Results with 1 hour of repair time Experiment 2 132
D.52 Ranking of policies for all weight combinations with an allowed weight deviation of 1 . . 133
D.53 Ranking of policies for all weight combinations with an allowed weight deviation of 2 . . 133
D.54 Ranking of policies for all weight combinations with an allowed weight deviation of 4 . . 133

E.1 Possible simulation settings 136
E.2 Possible crane settings 136
E.3 Possible rack configuration settings 137
E.4 Possible general rack settings 137

Contents

1 Introduction **1**
1.1 Background . . . 1
1.2 Jumbo Supermarkets . . . 1
1.3 Problem Definition . . . 2
1.4 Research Questions . . . 3
1.5 Methodology . . . 3
1.6 Report Structure . . . 6

2 Literature **7**
2.1 History and evolution of AS/RS . . . 7
2.2 Research topics . . . 8
2.3 Modelling AS/RS . . . 11
2.4 Literature gaps . . . 13

3 Problem definition **15**

4 Policies **17**
4.1 One of two cranes within an AS/RS is down . . . 17
4.2 One full AS/RS is down . . . 19
4.3 Performance evaluation . . . 20

5 Simulation Model **23**
5.1 Model requirements . . . 23
5.2 Modelling method . . . 24
5.3 Input data . . . 24
5.4 Model structure . . . 24
5.5 Assumptions and simplifications . . . 30
5.6 Options . . . 30
5.7 Output . . . 31
5.8 Model reusability . . . 32

6 Case Study **33**
6.1 Delivery Schedule . . . 33
6.2 Warehouse Flow . . . 36
6.3 Assumptions and simplifications . . . 39
6.4 System Logic . . . 39
6.5 Performance . . . 41
6.6 Current policies . . . 41

7 Verification & Validation **43**
7.1 Verification . . . 43
7.2 Validation . . . 49

8 Experiments **51**
8.1 Setup . . . 51
8.2 Scenarios . . . 54
8.3 Alternative configurations . . . 55

9 Results **57**
9.1 Scoring . . . 57
9.2 Experiment 1: One of two cranes within an AS/RS down . . . 57
9.3 Experiment 2: Both cranes within an AS/RS down . . . 61
9.4 Experiment comparison . . . 65
9.5 Sensitivity analysis KPI weights . . . 66
9.6 Additional observations . . . 67

10 Discussion **69**
10.1 Limitations . . . 69
10.2 Generalisation . . . 70

11 Conclusion **71**
11.1 One of two cranes within an AS/RS down. . . . 71
11.2 Both cranes within an AS/RS down . . . 72
11.3 Experiment comparison . . . 73
11.4 Contribution . . . 73
11.5 Future work . . . 73

A Research paper **77**

B Pseudo code **99**
B.1 Input data processing . . . 99
B.2 Pallet Generator . . . 99
B.3 Pallet Maker . . . 100
B.4 Crane . . . 101
B.5 Loader. . . . 106
B.6 Pallet returner. . . . 106
B.7 Results generator. . . . 107

C Simulation settings **109**

D Quantitative results **113**
D.1 No downtime . . . 113
D.2 Experiment 1: One crane down . . . 114
D.3 Experiment 2: Both cranes down . . . 123
D.4 1-hour downtime . . . 132
D.5 Sensitivity analysis KPI weights . . . 133

E Model Manual **135**
E.1 Input data . . . 135
E.2 Parameters . . . 136
E.3 Output . . . 138
E.4 Adjusting the model . . . 138
E.5 Additional files . . . 138

1 Introduction

1.1. Background

In the continuously evolving landscape of logistics, warehouses have gotten increasingly automated to reduce costs, reduce errors and improve efficiency. One of the innovations contributing to this is the Automated Storage and Retrieval System (AS/RS). AS/RS have been around since the mid-20th century and have developed a lot since then. At their introduction around the 1950s and 1960s, they were focused on simple storage and retrieval tasks using automated cranes and conveyors. Later, around the 1970s and 1980s, AS/RS became popular in manufacturing and distribution centres. Their performance improved due to technological advancements which led to more sophisticated designs and the usage of robotics and advanced software.

Nowadays, AS/RS are a vital component of modern warehousing and logistics making use of innovations such as real-time data analytics, machine learning, and integration with other smart technologies. As the name suggests, AS/RS automate the storage and retrieval of goods, or stock-keeping units (SKU). Various types of AS/RS exist, but in general, they consist of one or multiple racks in which SKUs can be stored, input- and output points and a crane or shuttle which moves the SKUs between the inputs, outputs and rack.

Some of the advantages of using AS/RS include increased efficiency, reduced labour costs, fewer errors and the possibility of 24/7 operation compared to manual systems. Also, they have a low space utilisation due to the high-density storage in the racks. Disadvantages include the high initial costs, complex maintenance of the system and limited suitability for goods with an irregular shape or size or with special handling requirements.

1.2. Jumbo Supermarkets

This research is conducted in collaboration with Jumbo Supermarkets, who provided a case study in which to examine the influence of partial downtime on AS/RS systems and how to minimise these effects. Jumbo Supermarkets is the second-largest supermarket in the Netherlands. Their supply chain consists of National/Central Distribution Centres (NDC/CDC) and Regional Distribution Centres (RDC) to supply fresh, frozen and dry groceries to and return packaging from stores. Next to that, E-Fulfillment Centres (EFC) and Hubs are used for home-delivery of groceries as can be seen in Figure 1.1.

Previously, Jumbo had an NDC for both fresh and dry groceries in Veghel, however, a few years ago, Jumbo opened the CDC KW for dry goods in Nieuwegein replacing the NDC for dry goods in Veghel. This new CDC is highly automated and delivers filled rollcages either directly to stores or to EFCs or RDCs which then deliver it to the stores of Jumbo. In 2024, Jumbo opened another CDC next to CDC KW for fresh products, replacing the last NDC in Veghel. This has a similar function to CDC KW, apart from the fact that the mechanisation works slightly differently. This difference is partly in the picking systems, and partly in an added Order Consolidation Buffer (OCB) which consists of four AS/RS for container storage instead of storing containers on the floor as in CDC KW. It is in their interest to investigate some of the challenges and resolutions in operating multiple parallel AS/RS.

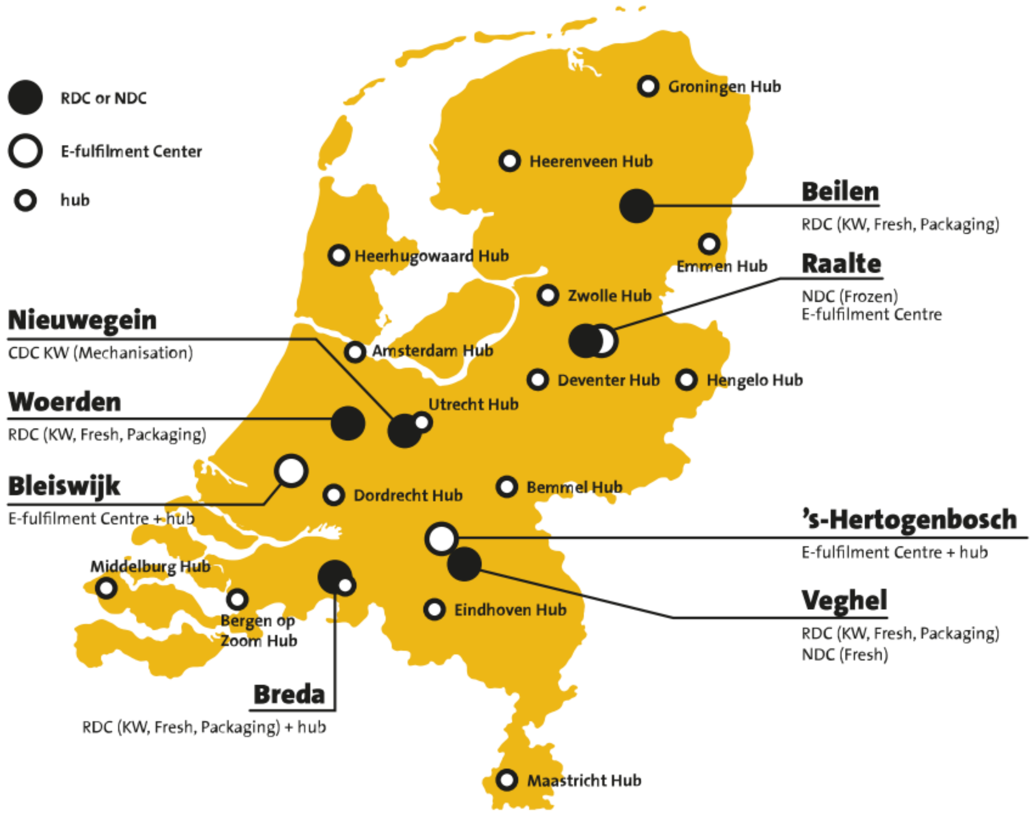


Figure 1.1: Supply Chain locations of Jumbo Supermarkets (Jumbo Supermarkets, 2022)

1.3. Problem Definition

Since AS/RS have been around for so long, they also have been researched a lot. However, the majority of AS/RS research concerns theoretical topics, which results in a limited impact on practice. AS/RS are mainly studied in isolation and the influence of upstream and downstream processes, and thus the influence of the total warehouse process, are not considered (Roodbergen & Vis, 2009)(Gagliardi et al., 2012a)(Azadeh et al., 2019). For example, the design and control of a single AS/RS in isolation has been well-researched, but the best way to deploy one or multiple AS/RS in the total warehouse system, and some of the practical challenges they bring, have not. One topic that has not been researched at all is partial downtime. This can occur in a redundant system with multiple parallel AS/RS or with multiple cranes. A system is partially down if part of the multiple AS/RS does not function or part of the multiple cranes does not function anymore. This could be due to unforeseen breakdowns of components, or due to scheduled maintenance. It is of relevance to research what the impact of this is and whether there are operational policies that can minimise this impact. Additionally, there is a lack of modelling frameworks designed to examine multiple AS/RS simultaneously. Another challenge in AS/RS research is the duplication of effort, with many researchers independently constructing their own models from the ground up.

The AS/RS in the OCB of Jumbo are in parallel. Their inputs are connected by a central conveyor, but their outputs are disconnected. While operating the warehouse, decisions have to be made about the workload distribution among the AS/RS, and whether to stop both cranes to aid the repair of a broken crane, or to continue operation of the other crane, slowing down repairs of the broken crane. Under normal circumstances, the workload is spread out over the four buffers. But with partial downtime, for example when one of the two cranes in a buffer or a full buffer is down, Jumbo could choose to alter this division of SKUs and thus workload over the buffer. However, they do not have previous experience with this and are curious about what the best policies are in the case of partial downtime of the buffers.

The goals of this research are as follows:

- Fill the research gap regarding the effects of partial downtime on AS/RS and OCB
- Develop a modelling framework for multiple parallel AS/RS which can be reused by other researchers or Jumbo to research other scenarios
- Advise Jumbo regarding how to deal with partial downtime in their OCB

1.4. Research Questions

The problem definition and research goals are represented by the main research question:

What is the best operational policy to minimise performance losses while operating parallel AS/RS under partial downtime of the system?

To answer this main research question and achieve the research goals, the following sub-questions are defined:

1. What is the current state of AS/RS research?
2. Which policies can be investigated to minimise the impact of partial downtime?
3. How to model parallel AS/RS?
4. How can the developed model be used elsewhere in AS/RS research?
5. How do the operational policies impact the performance of the system at Jumbo?

1.5. Methodology

The DEGREE problem-solving methodology by Rossetti will be used (Rossetti, 2015). The steps to follow in this methodology are as follows:

1. **D**efine the problem
2. **E**stablish measures of performance for evaluation
3. **G**enerate alternative solutions
4. **R**ank alternative solutions
5. **E**valuate and Iterate during process
6. **E**xecute and evaluate the solution

Executing the first step makes sure that the right problem is solved. Performing the second step ensures that the right metrics are used to examine the performance of the system under study. With steps 3 and 4, it is ensured that the right solution to the problem is developed. In step 5, it is evaluated how the process is going and the process can be reiterated until the desired level of modeling accuracy has been achieved. Lastly, step 6 is used to execute the solution and follow up to ensure that the solution works as expected, if possible.

This methodology was slightly altered by Rossetti for applications where simulation is involved, which is the case in this research for reasons explained in chapter 5. These phases are as follows and are visually represented in figure 1.2:

1. Problem Formulation
    (a) Define the problem
    (b) Define the system
    (c) Establish performance metrics
    (d) Build conceptual model
    (e) Document model assumptions
2. Simulation Model Building
    (a) Model translation
    (b) Input data modelling
    (c) Verification
    (d) Validation
3. Experimental Design and Analysis
    (a) Preliminary runs
    (b) Final experiments
    (c) Analysis of results
4. Evaluate and Iterate
    (a) Documentation
    (b) Model manual
    (c) User manual
5. Implementation

The first phase, problem formulation, covers the basics of the first two steps in the DEGREE process. The problem is formulated in collaboration with Jumbo and TU Delft to make sure it is useful for the company, but also academically contributing. The system is defined and it is made sure that the study focuses on the area of interest appropriately. The Key Performance Indicators (KPI), and thus the desired outputs of the model, will be identified and the conceptual model of the system will be built. Also, the model assumptions and simplifications will be documented which is important to be able to examine their effect on the model.

In the second phase, model building, one focuses on the main part of step 3 in the DEGREE process. The approach taken here is to start with a simple model and continuously expand until the desired complexity is reached. The complexity of the model must stay proportional to the degree of validity necessary for the study objectives and to the quality of the data. Input data is prepared in such a way that it is suitable for the model and specifications of the system are acquired. The model will then be verified to make sure that the model works as intended. This will be done by unit tests to test all the individual components of the code and with a sensitivity analysis where the system is observed while varying factors in the system and checking if the behaviour is as anticipated. Because the system at Jumbo is not operational yet and no real-world data exists, validation will be done by among other things, discussing the model with experts on the real system to confirm that the model represents the real system accurately enough.

After that, in the third phase, experimental design and analysis, some aspects from steps 3 and 4 of the DEGREE process are included. The preliminary runs are used to set the statistical parameters and to acquire a benchmark of the outputs of the model in the base scenario. After that, the different policies can be applied to the model and the corresponding outputs can be extracted.

The fourth phase, evaluate and iterate, is about going through things again. This should be done if one is not satisfied with the results of the simulation. Then it should be determined what else is needed to achieve the modelling objectives. This could be additional data, models, experimentation or analysis.

Finally, the last two phases, documentation and implementation, finish up the simulation process. When the simulation objectives have been achieved, the best-performing policies and simulation results should be documented from which conclusions can be drawn. Additionally, the model should be developed in such a way that the project can be easily modified or reused for other research purposes. Also, a user manual should be written so that non-analysts also know how to use the model.

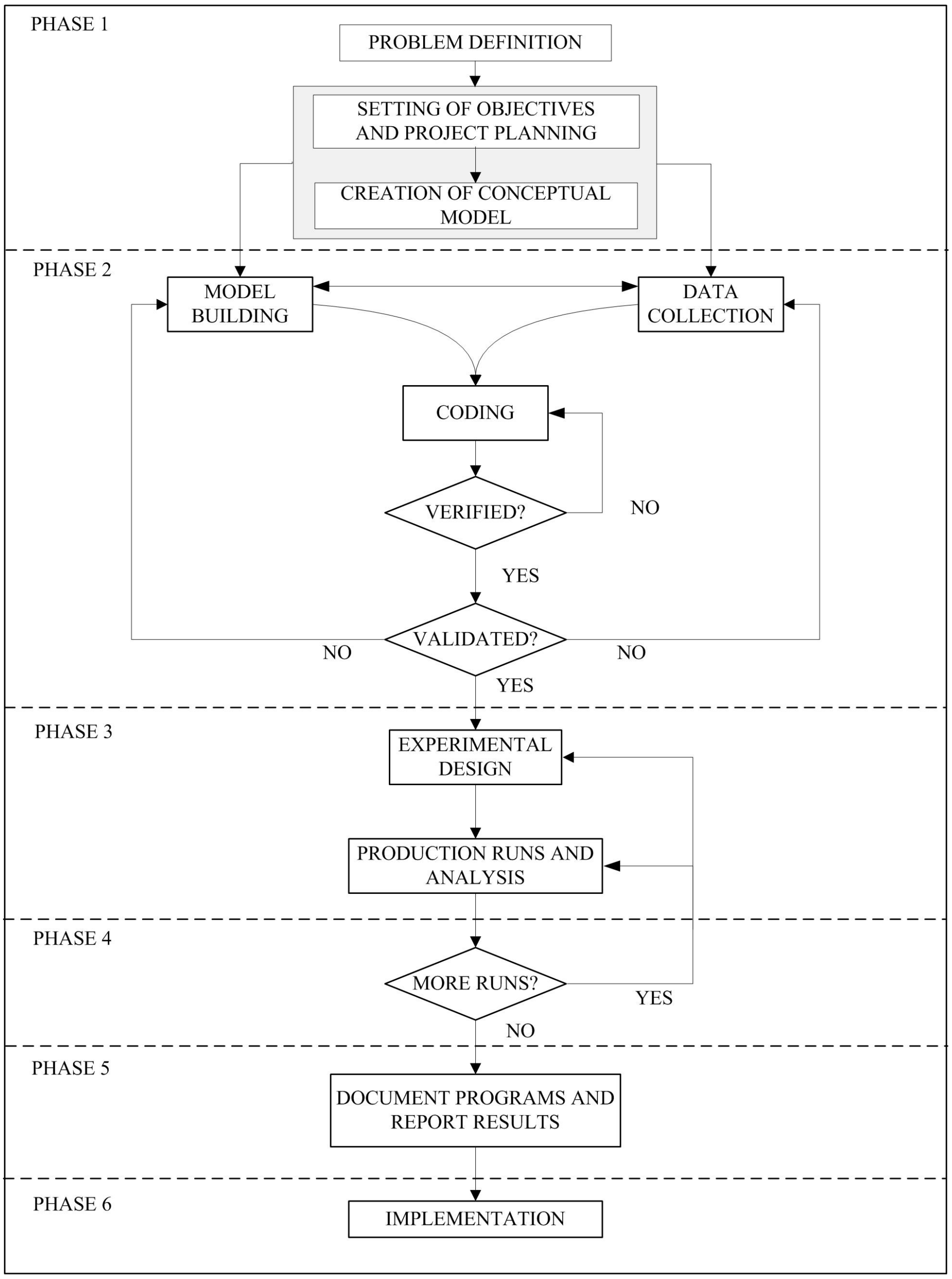


Figure 1.2: DEGREE methodology for simulation projects (Rossetti, 2015)

1.6. Report Structure

The report is structured as follows. In chapter 2, existing literature on AS/RS research related to the topic of this research will be discussed. This chapter therefore also answers sub-question 1.

Chapter 3 defines the problem that will be studied in this research. In chapter 4, different strategies for minimising the effects of partial downtime will be designed and formed into operational policies. Here, sub-question 2 will be answered.

In chapter 5, first, the development of the generic simulation model will be discussed. After that, it will be explained how the conceptual model was translated into code and how the model can be used to study other systems. These chapters therefore answer sub-questions 3 and 4.

Chapter 6 defines the system at Jumbo, explains its implementation into the simulation model and discusses the performance metrics of the system. Chapter 7 will explain how the model and the implementation of the system under study were verified and validated. In chapter 8, the experiments will be explained and in chapter 9 the results of the simulations will be presented, thereby answering sub-question 5.

The accuracy of the results will be discussed in chapter 10 and conclusions will be drawn in chapter 11, which also answers the main research question.

Overall, chapters 1 to 4 represent phase 1 of the methodology. Chapters 5 to 7 represent phase 2. Finally, chapters 8 and 9 represent phase 3, chapter 10 represents phase 4 and chapter 11 represents phase 5. Phase 6, implementation, is not possible for this research because the system is not fully operational yet, and because this is outside of the scope of this research.

2

Literature

AS/RS research has been going on for a long time with initial developments dating back to the 1950s. Since then, a lot of papers concerning the design and operation of these systems have been published. However, there are still plenty of research topics concerning AS/RS that are yet to be explored. This section will discuss the existing AS/RS literature and identify research gaps, thereby answering sub-question 1.

2.1. History and evolution of AS/RS

The evolution of AS/RS throughout history has been coupled with technological innovations and evolving industrial needs. In the 1950s and 1960s, the main focus was on automated material handling in general in warehouses and distribution centres.

One of the first systems of this kind was installed at an IBM warehouse (Witt, 1974). Their main motivation for installing an AS/RS was the reduced needed land area and savings on manpower operating costs compared to manual systems. A computer simulation was used to aid the design of the system where the goal was to minimise the overall costs while taking into account the required storage volume and throughput requirements as can be seen in figure 2.1. The results of this simulation yielded an optimal height, length and number of aisles.

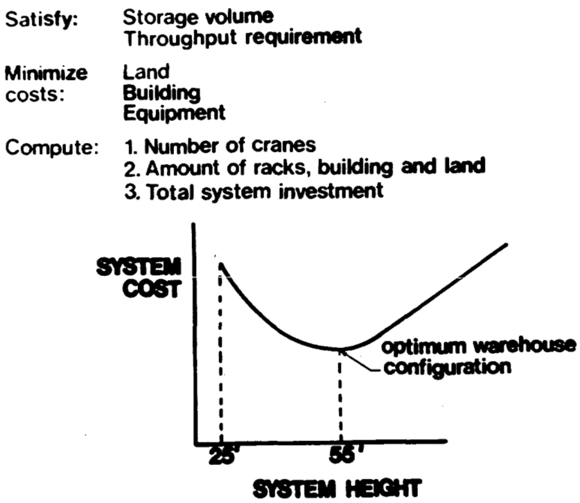


Figure 2.1: Graph of AS/RS model results (Witt, 1974)

Later, during the 1970s and 1980s, several different types of AS/RS were developed. Examples of this are mini-load AS/RS and AGV-based AS/RS. During these decades, the focus shifted towards improving the efficiency and accuracy of these systems.

As an example, Hausman et al. investigated optimal storage assignment in a mini-load AS/RS by comparing random storage assignment, full turnover-based assignment and class-based assignment (Hausman et al., 1976). In a mini-load AS/RS, bins or trays containing smaller items are stored in the racks. Smaller amounts of items are picked from the bin or tray after which they are returned to the rack. Therefore, the location assignment of these bins or trays can greatly impact the efficiency of the system. Their conclusion was that turnover-based storage assignment results in a significant reduction in crane travel times and therefore increases the throughput of the system.

After this, during the 1990s, newly developed technologies began to get incorporated into AS/RS. Innovation in the domains of among others robotics, computer control systems and RFID further enhanced their capabilities.

For example, Lee et al. take advantage of these innovations by investigating sequencing methods for AS/RS (Lee & Schaefer, 1997). In their paper, they try to find an optimal method to determine the order in which retrieval requests are performed. By applying a dynamic heuristic, they managed to improve the throughput of the system.

Eventually, from the 2000s to the present, further technological developments shaped AS/RS into what they are now. Robotics and sophisticated control systems enable greater flexibility and adaptability. The current focus is on energy efficiency, integrating Industry 4.0 and possibly incorporating artificial intelligence, machine learning and predictive maintenance.

2.2. Research topics

Bertolini et al. performed a bibliometric analysis to map the evolution of research themes within a field by studying over a thousand papers (Bertolini et al., 2023). By doing this, keywords can be categorised into the categories shown in figure 2.2. This section will highlight research concerning some of the current research topics.

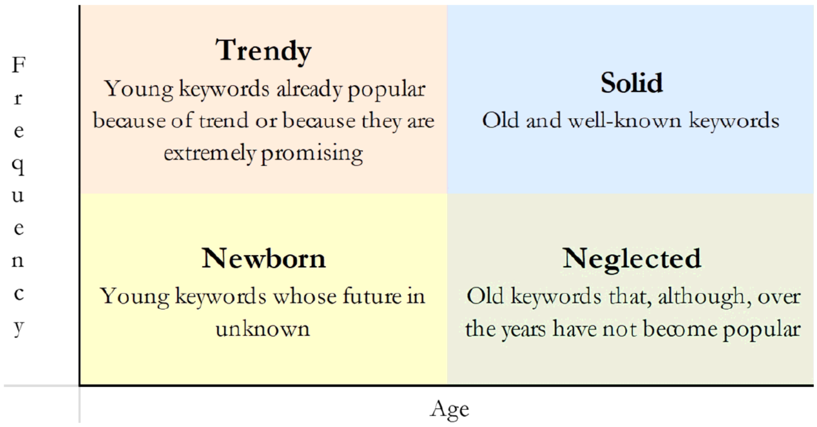


Figure 2.2: Keywords classification matrix (Bertolini et al., 2023)

According to their study, trendy topics include shuttle-based storage and retrieval systems (SBS/RS) and multi-deep AS/RS. Newborn topics include multiple input/output (I/O) points and energy consumption. Solid topics include storage allocation, job scheduling and the physical design of the system.

2.2.1. SBS/RS

(Borovinšek et al., 2017) propose a multi-objective optimisation approach for designing Shuttle-Based Storage and Retrieval Systems (SBS/RS) with a focus on minimising average cycle time, energy consumption, and total investment cost. Seven design variables, such as the number of aisles and velocities, are considered. The Non-Dominated Sorting Genetic Algorithm II (NSGA II) is used due to the non-linear nature of the objective function. Pareto optimal solutions are sought for efficient and flexible decision-making in warehouse design at the early planning stage.

(Lerher et al., 2015) present an analytical travel time model for SBS/RS. The model considers the elevator lifting table and shuttle carrier operating characteristics, incorporating factors like acceleration, deceleration, and maximum velocity. Mean travel time is calculated using probability theory and assuming uniformly distributed storage rack locations. The proposed analytical model is compared with approximation and simulation models, demonstrating consistent correlation with simulation results across various SBS/RS types.

(Lerher, 2017) employs Design of Experiment (DOE) to identify factors influencing the performance of SBS/RS. After determining these factors, DOE analysis is used to optimise SBS/RS throughput performance. The study shows that the number of columns, shuttle carrier velocity and acceleration, and the elevator's lifting table velocity and acceleration interactively affect various performance measures, including cycle times and throughput performances. The factors studied include average single and dual command cycle times, as well as throughput performances for both the shuttle carrier and elevator's lifting table, along with warehouse volume.

2.2.2. Multi-deep AS/RS

(Yang et al., 2015) focus on optimising storage rack design in a multi-deep compact AS/RS, accounting for S/R machine acceleration and deceleration. The study presents general models to determine the optimal ratio of dimensions, minimising travel time for various speed profiles. The impact of S/R machine speed profiles and fixed dimensions are examined through numerical experiments.

(Yu & De Koster, 2009) investigates the optimisation of layout for compact, multi-deep AS/RS. The research aims to minimise the expected cycle time under various storage policies. The expected single-command cycle time for a full-turnover-based storage policy is derived and a model to determine optimal rack dimensions is proposed. By simplifying the model, optimal dimensions are analytically determined based on rack capacity and ABC curve skewness, resulting in significant cycle time reduction compared to random storage policies.

2.2.3. Multiple I/O

(Song & Mu, 2022) perform an operational optimisation in a multiple-input/output (multi-I/O) points AS/RS. The system involves twin stackers operating simultaneously in the same aisle without crossing each other with the I/O point being unknown. Key findings include the significant efficiency improvement of storage and retrieval with double unit transport, particularly during peak times. The proposed method outperforms phased optimisation, showcasing a nearly 30% reduction in makespan for large-sized examples.

(Tanaka & Araki, 2009) study the routing problem in unit-load AS/RS with separate input and output points, focusing on the shared storage policy. Their goal is to optimise the travel route of an S/R machine for given storage and retrieval requests, minimising total travel time. They formulate the problem using 0–1 integer linear programming and two types of dwell point settings are considered: it is either the input or output point. An exact solution algorithm using a general Mixed Integer Linear Programming (MILP) solver is proposed and demonstrated through numerical experiments.

2.2.4. Energy consumption

(Lerher et al., 2014) introduce an energy efficiency model for mini-load AS/RS. Their proposed model emphasises the importance of considering energy and environmental factors in the design process. By reducing energy consumption and therefore CO_2 emissions, their model aims to contribute positively to both economic and environmental aspects. The main findings were that extreme velocity profiles also generate extreme power demands.

(Meneghetti et al., 2015) introduce new factors in their study such as unit load weight and differentiation of shifts along the horizontal and vertical axes for accurate energy calculation. An optimisation model combining Constraint Programming with Large Neighborhood Search includes both time and energy-based objectives, considering multiple weight unit loads and energy recovery. Their results state that regardless of the demand curve and optimisation objective, intermediate height rack shapes achieve the best energy efficiency while lower ones have better travel time performance.

(Meneghetti & Monti, 2013) also compare a time-based full turnover strategy with an energy-based strategy in an AS/RS, coupling each rack location with crane energy consumption. Several crane models are considered, analysing assignments based on dedicated zone shapes, time and energy performances. Various rack shapes and product ABC curves are examined, and dwell-point policies are evaluated from a sustainable perspective, combining energy-saving goals with the traditional aim of picking time reduction.

2.2.5. Storage allocation

(Gagliardi et al., 2012b) emphasize the importance of assessing the applicability of assumptions to real industrial settings, unlike many theoretical works relying on tight assumptions. A discrete-event simulator is introduced, replicating an industrial AS/RS in the food industry. Using real data scenarios, various storage assignment policies are compared. The experimental results highlight the deviation of system behaviour from theoretical expectations when realistic conditions are considered.

(Hsieh & Tsai, 2001) investigate the impact of a storage assignment policy aligned with manufacturing needs on the performance of an AS/RS and the overall production system. The study proposes a bill of material-oriented (BOM) class-based storage assignment method. A case study demonstrated the effectiveness of the proposed method, comparing it with a random storage assignment approach. Results showed the efficiency of the BOM-oriented class-based AS/RS assignment method.

(Kulturel et al., 1999) use computer simulations to compare two shared storage assignment policies. The main performance measure is the average travel time of the AS/RS for storing and retrieving products. The study investigates system sensitivity to product variety, inventory replenishment lead time, demand rate, inventory policy, and product classification technique. Their conclusion is that the turnover-based policy tends to outperform the duration of stay-based policy, but the difference becomes insignificant under specific conditions.

2.2.6. Job scheduling

(Wang & Yih, 1997) use artificial neural networks trained on simulation results from various experimental designs to design a control system. The neural network takes system configuration and required performance levels as inputs and provides control strategies for storage location assignment, retrieval location selection, queue selection, and job sequencing as outputs. The study explores different neural network topologies and training parameters, demonstrating the feasibility of the proposed approach with an 84% accuracy in identifying novel data.

(Ekren & Arslan, 2024) aim to improve the performance of an SBS/RS through a machine learning (ML) approach. An SBS/RS is designed which enables shuttles to travel between tiers. To address the operational complexity of shuttles in this design, an ML-based algorithm for job selection is implemented. Their ML-based solution is compared with traditional scheduling approaches, such as first-in-first-out and shortest process time scheduling rules. The findings demonstrate that in most cases the Q-learning approach outperforms the two static scheduling approaches.

(Elsayed & Unal, 1989) introduce heuristics and analytical models for addressing the order batching problem, focusing on minimising total travel time. Four heuristics are developed combining orders into a single tour. Additionally, an analytical model is presented to estimate the travel time of the S/R machine, considering the number of locations to be visited and the physical specifications of the structure. The paper also provides expressions for upper and lower bounds for travel time.

2.2.7. Physical system design

(Z. Chen et al., 2015) introduce the concept of a bi-directional flow-rack (BFR), deviating from traditional flow-racks by designing bins in adjacent columns to slope in opposite directions. In a BFR, unit loads are stored on one side and retrieved from the other, allowing for dual-command (DC) operations on both sides. The paper presents a travel time model for BFR systems, establishing throughput baselines for different configurations. A DC operation generation method is proposed and evaluated through simulation experiments, comparing the throughput performance of BFR and traditional single-directional flow-rack (SFR) systems.

(Malmborg, 2001) propose a modification to a well-known rule of thumb for evaluating storage rack configurations in AS/RS. Their modification eliminates the need for assumptions regarding the proportion of single and dual command order picking cycles and the total storage capacity requirements when comparing randomised versus dedicated storage. The modified rules of thumb are designed to ensure computational efficiency for analysing a broad range of rack design alternatives in large-scale applications.

2.3. Modelling AS/RS

To study AS/RS, generally, two types of models can be used, either an analytical model or a simulation model. The main types of analytical and simulation models used in AS/RS research will be briefly discussed in this section.

2.3.1. Analytical models

Analytical models are typically used for quick analysis of a system because they take a relatively short time to develop and do not require much computational power. This can be useful for preliminary estimations or for studying a large number of different system configurations. They are, however, limited in their ability to model complex systems.

Queuing networks
Queuing Networks are a stochastic way of modelling in which different variations exist such as Open, Semi-Open and Closed Queuing Networks. A queuing network is a collection of servers, representing the resources of the system, and customers competing for those resources where they possibly have to wait in a queue for those resources. The goal of analysing queuing networks is to determine performance measures such as the number of customers in the system or queue, average time spent in the system or queue and system utilisation factor. The main reason for using queuing networks is the relatively high accuracy and efficient model evaluation (Bernardo & Hillston, 2007).

For example, (Ekren et al., 2014) use a Semi-Open Queuing Network (SOQN) to estimate key performance measures for an Autonomous Vehicle Storage and Retrieval System (AVS/RS). In the context of an AVS/RS, the jobs correspond to S/R transaction requests, and the autonomous vehicles (AVs) represent the servers. Modelling the AVS/RS as an SOQN accounts for potential wait times between AVs and S/R transactions. The queuing network is constructed by establishing general travel times for pre-defined servers. The AVS/RS system is conceptualised as a single-class, multiple-server SOQN. Employing the Modified Gordon-Newell Method (MGM), the network is solved, and key performance measures are derived. The MGM technique is applied to a warehouse in France utilising AVS/RS in their operations.

Mathematical programming
Mathematical programming is known as the part of operations research that researches the optimal allocation of resources between competing activities (Jensen & Bard, 2002). This way of modelling is deterministic and generally exists of an objective function and a set of constraints with which an optimal solution is desired to be found.

For example, (Man et al., 2021) study a bi-objective optimisation problem for an AS/RS in a container terminal with a task release time and due date using bi-objective mixed integer linear programming. Their goal is to minimise the crane travel time and total tardiness simultaneously. Several algorithms to solve this problem are developed and their efficiencies are compared.

2.3.2. Simulation models

Simulation models cost more time to develop, but they are suitable to study complex AS/RS configurations yielding realistic results because constraints do not have to be used. Very specific operational policies or system designs can be modelled with this. Also, it allows for easier understanding of system behaviour.

Discrete Event Simulation
Discrete Event Simulation (DES) is one of the most popular modelling techniques which has been greatly developed over time (Robinson, 2005). The technique models systems as a sequence of events occurring at discrete moments in time. These events can change the state of the system, or add more events to the events list. Between the events, no changes happen, such that the system jumps in time from event to event until the stop conditions of the simulation have been met.

For example, (Marolt et al., 2022) study relocation and storage assignment strategies in a multi-deep AVS/RS using DES. Relocation is necessary for multi-deep AS/RS if an SKU is blocking another SKU behind it that needs to be retrieved. This occurs when different items are stored in the same lane and the FIFO principle is followed. A widely used kinematic model was adopted to calculate travel times. The three different strategies investigated are a random strategy which chooses random locations, a nearest neighbour (NN) strategy relocating in the nearest available location, and a depth-first (DF) strategy which chooses the most in-depth possible location and if there are multiple, it chooses the closest one. It was concluded that for lower fill grades, the NN relocation and NN storage performed best, and for higher fill grades the NN relocation and DF storage performed best.

Agent-Based Modelling
Agent-Based Modelling (ABM) simulates the interaction and actions of autonomous agents in an environment. It is a way of modelling that has been developed more recently compared to for example DES and it can be used to study more complex systems by focusing on the individual actions of agents. These models are built bottom-up by identifying agents, defining their behaviour, establishing connections between them and setting environmental variables (Macal & North, 2005).

(Eroglu & Yetkin Ekren, 2022) study collision and deadlock prevention in an SBS/RS with an agent-based modelling approach. Three types of agents are defined, a demand agent, a shuttle agent and a deadlock control agent which can each make decisions independently. The demand agent keeps track of storage and retrieval requests and calculates the distance to perform those actions for the different shuttles. The shuttle agent decides which shuttle gets which task, determines the dwell point and triggers policies in case of possible collisions or deadlock. The deadlock control agent is triggered in case of a possible deadlock and prevents the situation. The proposed system is tested against a first-come-first-served policy and shortest processing time policy for transaction selection and the proposed system performed best in terms of average flow time per transaction and total number of transactions processed for 1, 2 or 3 shuttles per tier.

Hybrid models
Hybrid simulation can be described as a modelling method that combines at least two different methods. These methods are mostly system dynamics (SD), DES or ABM, however, a hybrid model can also exist as a combination of an analytical and simulative method. Using a hybrid model is still new but has become quite popular in the past 2 decades (Brailsford et al., 2019).

(Barbato et al., 2019) evaluate policies for AS/RS by using a model combining discrete events and agent-based paradigms. A four-step methodology is introduced. The first step is to make a descriptive model of the AS/RS and characterise it using entities, events and activities. Entities are then modelled as agents while the order composition of single operators can be modelled using a discrete event approach. The next step was to validate the model using a data set from an industrial partner and to analyse bottlenecks. The third step was to optimise the bottleneck by implementing a policy. To accommodate this policy, it is assumed all orders for the day are known in advance. The last step was to evaluate the new policy using the simulation tool. The advantage of their framework turned out to be that it could be more effectively encoded by optimisation models.

2.4. Literature gaps

The research topics and papers mentioned in this chapter represent a fraction of all literature that exists on AS/RS. However, there are still areas that have not been explored at all or not well enough. The majority of AS/RS research seems to study the design and operation of AS/RS in isolation. In practice, AS/RS are often coupled to other systems upstream and downstream of the AS/RS itself, which can in turn also be other AS/RS systems. These upstream and downstream processes influence the performance of the AS/RS because they dictate the flow of products into and out of the AS/RS. Therefore, to accurately study the operation and design of AS/RS, these factors should be taken into account.

A topic which is also absent in AS/RS literature is downtime. This is not surprising when they are mainly studied in isolation because when an isolated AS/RS is down, there is not much to study. However, in production systems or warehouses where multiple AS/RS are working together, such as in Order Consolidation Buffers, if one of them is down for a period of time, different strategies can be employed. These strategies aim to minimise the overall impact on the system's performance. This is especially interesting in systems with multi-crane AS/RS where just one of the cranes can be down and it needs to be determined how much workload the remaining crane or cranes can handle.

Next to that, there is a large duplication of effort with most researchers building their own models from the ground up. Few modelling frameworks for AS/RS exist, and the ones that do are often limited in their flexibility to adapt to specific system configurations or circumstances. Currently, no framework exists where the influence of upstream and downstream processes can be taken into account, and where multiple AS/RS which are working together can be studied.

Table 2.1 summarizes some of the most relevant studies concerning the aforementioned topics. It can be concluded that downtime is a topic within AS/RS research that has been neglected. This is confirmed by (Bertolini et al., 2023) who studied over a thousand papers, and these topics do not come back as used keywords in AS/RS research in their bibliometric analysis.

In conclusion, the answer to sub-question 1 is that AS/RS research has been around for a long time and a large variety of topics have been researched. However, there are still topics that have not gotten any attention such as downtime in AS/RS. Also, the impact of research on industry has been limited since the majority of researchers focus on AS/RS in isolation and do not take practical aspects into account such as the upstream and downstream system influence.

	Focus		Method		Type			Characteristics							
Reference	Design	Operation	Analytical	Simulation	CB-AS/RS	SBS/RS	AVS/RS	Dual crane	Multi I/O	Multi deep	Downtime	Upstream process	Downstream process	Multiple AS/RS	Model reusable
Gagliardi et al., 2014	X	X		X	X				X	X		X			X
Turner, 2020		X			X	X	X			X		X	X		
Tappia et al., 2019		X	X		X	X							X		
Gogi and Gupta, 2023		X		X	X						X				
Lewczuk, 2021		X		X	X							X	X		
W. Chen et al., 2022	X		X		X			X	X						
Bansal et al., 2021		X	X		X								X		
Singbal and Adil, 2023	X	X		X	X										X
This research		X		X	X			X	X	X	X	X	X	X	X

Table 2.1: Papers studying topics related to this research

3 Problem definition

This chapter will describe the generic problem that will be studied in this research.

In the literature review, it was observed that topics such as partial downtime in AS/RS and the operation of parallel AS/RS have not gotten any previous attention. To provide the industry with guidance on how to deal with this, this topic will be studied in this research. Partial downtime can occur in redundant systems with multiple AS/RS in parallel, possibly with multiple cranes. There are several decisions to make while operating such systems under partial downtime.

The more complex the system, the more possible solutions exist to mitigate performance losses during partial downtime. To examine a larger range of possible solutions, the following system characteristics should apply:

1. The inputs of the parallel AS/RS should be connected allowing incoming goods for the upstream systems to be distributed to any AS/RS. This could for example be represented by a centralised conveyor that is connected to the inputs of all AS/RS.

2. The system should consist of at least three parallel AS/RS. This allows for a decision about whether to move the workload of an AS/RS that broke down to the neighbouring AS/RS, which might be desirable in case goods need to stay close to their original destination, or to spread it out across all other AS/RS.

3. The AS/RS should have multiple cranes. This allows for the possibility of just one crane breaking down and continuing operation with the remaining crane, possibly with a reduced service area depending on the breakdown location of the crane. If the cranes operate on the same rail, decisions might need to be made about whether to stop the remaining crane to speed up repairs or not.

4. The outputs of the parallel AS/RS should be connected allowing outgoing goods to the downstream systems to be distributed from any AS/RS.

5. There should be an option to directly unload goods from the AS/RS and remove them from the system. This could be necessary in case the extra workload cannot be handled by the other AS/RS. In this scenario, these directly unloaded goods need to be manually handled, stored somewhere and manually retrieved later.

It turns out that systems with these characteristics often exist in the industry in the form of Order Consolidation Buffers. In such systems, manufactured goods for orders are temporarily stored before being shipped once the order is complete. An illustration of such a system with the mentioned characteristics can be seen in figure 3.1.

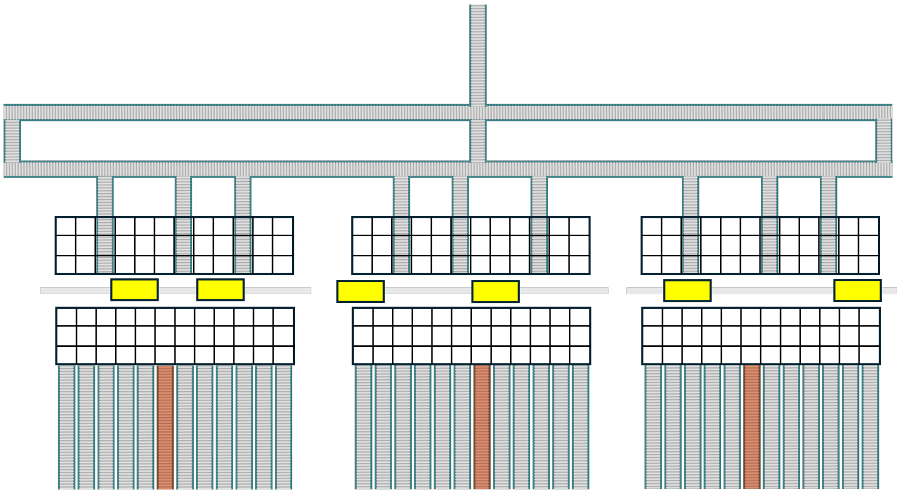


Figure 3.1: Illustration of parallel AS/RS with specified characteristics

Overall, to mitigate performance losses during partial downtime in such a system, guidance on the following decisions is desired:

- One crane within an AS/RS breaks down:
  - Should the operation of this AS/RS be continued with the remaining crane, possibly slowing down repairs?
  - Should part of the workload for this AS/RS be redistributed to the other AS/RS?
- Full AS/RS breaks down:
  - How should the workload originally meant for this AS/RS be redistributed?

4 Policies

This chapter will explain the developed and investigated policies to minimise the impact of partial downtime and thereby answer sub-question 2.

To be able to study partial downtime in AS/RS, the system under study must have specific characteristics. It should consist of at least two parallel AS/RS, preferably more. Optionally, each AS/RS could have multiple cranes so that the effects of downtime of just one of the cranes can be studied. Also, the inputs of the AS/RS should be connected, for example via a central conveyor, and the outputs should be disconnected, which makes the distinction between parallel AS/RS and multi-aisle AS/RS.

In this research, a system consisting of four parallel AS/RS with two cranes each will be studied and it is assumed each AS/RS has an output point that can at all times be used to unload goods. This increases the range of different possible policies that can be evaluated. There are two scenarios that will be studied; one of the two cranes in an AS/RS is down, and one full AS/RS is down. In the end, the best-performing policies for both scenarios can be compared to draw a conclusion about whether a broken crane should be moved so that the other crane can continue operating, or if both cranes should be stopped, speeding up repairs.

4.1. One of two cranes within an AS/RS is down

In this scenario, the other crane is still functioning and could possibly take over the tasks of the other crane, depending on the workload. Choices must be made about what to do with goods destined for this AS/RS. The capacity of this AS/RS could be changed, this means that during the division of workload across the AS/RS, this AS/RS gets a smaller share of the overall workload.

Once a crane breaks down, it needs to be moved to the side so that the other crane, which operates on the same rail, can continue operation. The moving of the crane takes time, during which the other crane also needs to be stopped because of safety procedures. This time is added to the total downtime for this experiment. Pallets that arrive at the AS/RS inputs during this time need to be unloaded via the failure lane of the partner buffer to avoid congestion on the central conveyor. It should be noted that the crane with downtime later still blocks part of the rack which reduces the available working area of the other crane. Therefore, pallets that were already in production for outputs that are now blocked are also unloaded via the failure lane. An overview of selected policies can be seen in table 4.1.

Policy	AS/RS capacity
0	Keep at 100%
1	Lower to 75%
2	Lower to 67%
3	Lower to 50%
4	Lower to 0%

Table 4.1: Policies when one of the two cranes within an AS/RS is down

4.1.1. Policy 0 - 100% capacity

This is the benchmark scenario where the AS/RS capacity stays at 100%. With lower workloads, the remaining crane could possibly still handle the workload, with higher workloads, this could cause problems.

Advantages:

- Operators do not have to take any action
- Downstream systems keep the same workload
- Workload for other AS/RS not increased

Disadvantages:

- Workload for remaining crane could become too high causing delays or congestion

4.1.2. Policy 1 - 75% capacity

This policy slightly reduces the capacity of the AS/RS and therefore the workload it receives.

Advantages:

- Slightly reduced workload might make it so that the remaining crane can handle it

Disadvantages:

- Workload for downstream systems becomes slightly skewed

4.1.3. Policy 2 - 67% capacity

This policy reduces the workload the AS/RS receives slightly more.

Advantages:

- Reduced workload might make it so that the remaining crane can handle it

Disadvantages:

- Workload for downstream systems becomes skewed

4.1.4. Policy 3 - 50% capacity

In this policy, the workload is halved.

Advantages:

- More reduced workload might make it so that the remaining crane can handle it

Disadvantages:

- Workload for downstream systems becomes more skewed
- Workload for other AS/RS increased more

4.1.5. Policy 4 - 0% capacity

In this policy, the capacity is fully reduced to 0%. The remaining crane is only used to handle pallets that were already in production for this buffer.

Advantages:

- Fully reduced workload might make it so that the remaining crane can handle it
- Fewer pallets stuck in buffer compared to also stopping this crane

Disadvantages:

- Workload for downstream systems becomes even more skewed
- Workload for other AS/RS increased even more

4.2. One full AS/RS is down

In this scenario, one full AS/RS is down. This could happen when both cranes are broken, or when one crane is broken but the decision is made to stop both cranes in order to speed up repairs. Newly generated orders will avoid this now broken buffer, but a decision has to be made about what to do with goods that were already in production for the AS/RS which is now out of operation. Pallets could either be directly unloaded at another rack or can get a destination there. An overview of selected policies can be seen in table 4.2. Each policy will be explained in the following subsections.

Policy	New destination	Operation
0	No change	Wait on conveyor
1	Neighbouring AS/RS	Directly unload at new AS/RS
2	Neighbouring AS/RS	Change destination and store in new AS/RS
3	Spread over all other AS/RS	Directly unload at new AS/RS
4	Spread over all other AS/RS	Change destination and store in new AS/RS

Table 4.2: Policies when both cranes down

4.2.1. Policy 0 - No action

This is the benchmark scenario where nothing is changed to which the performance of other policies can be compared. This policy could work in situations where the expected downtime is very short or there is a lot of buffer capacity at the inputs of the AS/RS.

Advantages:

- Does not require any additional work from the operators

Disadvantages:

- Will cause the central conveyor to overflow with goods for longer downtime durations

4.2.2. Policy 1 - Unload at partner

In this policy, all goods that were destined for this AS/RS are rerouted to the neighbouring AS/RS. The goods rerouted to the other AS/RS are unloaded directly there. Operators need to manually move and sort them before storing them on the floor.

Advantages:

- Storage capacity of other AS/RS not compromised
- Smaller distance to move relocated goods to original destination if necessary

Disadvantages:

- Causes added manual work
- Increased workload for cranes in other AS/RS

4.2.3. Policy 2 - Store in partner

In this policy, all goods that were destined for this AS/RS are rerouted to the neighbouring AS/RS. In this case, the destination of the goods that are rerouted to the other AS/RS is changed which means they can be stored there instead of having to unload them directly.

Advantages:

- No added manual work
- Smaller distance to move relocated goods to original destination if necessary

Disadvantages:

- Storage capacity of other AS/RS compromised
- Higher increased workload for cranes in other AS/RS

4.2.4. Policy 3 - Unload spread out

In this policy, all goods that were destined for this AS/RS are spread out over all other AS/RS. The goods rerouted to the other AS/RS are unloaded directly there. Operators need to manually move and sort them before storing them on the floor.

Advantages:

- Storage capacity of other AS/RS not compromised
- Slightly less increased workload for other AS/RS cranes
- Added manual work spread out more evenly over operators

Disadvantages:

- Causes added manual work
- Goods are stored further from their original destination

4.2.5. Policy 4 - Store spread out

In this policy, all goods that were destined for this AS/RS are spread out over all other AS/RS. The destination of the goods that are rerouted to the other AS/RS is changed which means they can be stored there instead of having to unload them directly.

Advantages:

- No added manual work
- Slightly less increased workload for other AS/RS cranes

Disadvantages:

- Storage capacity of other AS/RS compromised
- More work to replan rerouted goods when not done automatically by system

4.3. Performance evaluation

The overall performance of the system will be evaluated with a weighted sum of the KPIs seen in table 4.3 which also gives an explanation as to why each KPI is relevant. These weights were determined based on expert consultation.

KPI	Weight	Relevant data
Output delay	10	Aggregate trip delay [min]
		Number of trips with delay [trips]
Upstream system interference	8	Maximum amount of pallets on conveyor [pallets]
		Average amount of pallets on conveyor [pallets]
Added manual work	6	Total number of direct unloads [pallets]
		Maximum number of direct unloads for one rack [pallets]
Robustness	4	Maximum crane utilisation [%/h]
		Longest continuous utilisation [min]
		Maximum fill grade [%]
		Minimum margin between departure times per dock [min]
Resilience	2	Maximum average rack fill grade since downtime stop [%]
		Maximum average crane utilisation since downtime stop [%]

Table 4.3: Policy performance evaluation criteria

4.3.1. Output delay

This factor is most important because this affects the downstream processes of the system, which is the loading process in the case study. When goods arrive at the outputs too late, trips will be delayed which affects the clients. It was considered to include the maximum delay for 1 trip as a KPI, but it turned out that this was always influenced by the trips being stuck in a temporarily unreachable part of the rack which caused this value to be the same for all policies.

4.3.2. Upstream system interference

This is important because congestion at the inputs could cause upstream systems to have to come to a standstill. The parallel AS/RS are connected via a central conveyor leading to them. When this conveyor is congested with goods, the production systems have nowhere to leave the produced goods and have to come to a standstill.

4.3.3. Added manual work

When goods are directly unloaded, they have to be manually stored elsewhere and later retrieved again. This costs extra labour and therefore also introduces extra costs and should be minimised. Next to that, this extra storage space on the floor needs to be available in the first place and this manual handling increases the likelihood of errors.

4.3.4. Robustness

In this application, robustness is defined as the ability to maintain performance during further variations and disturbances, since downtime itself is already a disturbance. It is determined whether the system is pushing its limits under this policy and cannot handle further disturbances and variations, or if the system could handle those.

4.3.5. Resilience

In this application, resilience is defined as the ability to recover from downtime. To judge this, plots from several KPIs are assessed on the time it takes to return back to normal and the crane utilisation and fill grades after downtime are monitored. Together with robustness, as long as there are no delays, there is no upstream interference and there is no added manual work, making it less important if the policies score worse on this.

Scores for each KPI will be determined arithmetically on a scale from one to ten based on reference values. These scores are multiplied by the weight and summed to determine the overall score for each policy. A detailed explanation of the scoring will be given in chapter 9.

In conclusion, this chapter proposed a set of policies for two experiments to answer sub-question 2. In case one of two cranes in an AS/RS is down, the policies reduce the workload this AS/RS will receive with a varying reduction between 0% and 100%. In case both cranes in an AS/RS are down, the policies either store or directly unload the already generated production in the neighbouring AS/RS or spread out over the other AS/RS.

5 Simulation Model

This chapter will explain the model requirements, the motivation for the type of model chosen, how the model is built up and which assumptions are made. This answers sub-question 3. Additionally, it will explain how the model can be reused which answers sub-question 4.

5.1. Model requirements

The model should be to represent redundant systems with parallel AS/RS. This means that it should consist of components of which multiple instances can be created in the simulation environment. Essential components include data elements such as pallets and racks, as well as functional elements such as an order generator, upstream (production) system, crane, and downstream system.

The model should be capable of examining system performance across various scenarios, including different downtime start times, durations, and workloads. Furthermore, it should support the implementation of various operational policies altering system behaviour. Some processes will be stochastically modelled, causing the need to be able to run simulations with many replications with different random seeds for experiments. Additionally, the model should facilitate the extraction of the KPIs outlined in table 4.3. Finally, to help with understanding the system's behaviour and with debugging, there should be a possibility to output an event log and plots of KPIs throughout time.

To ensure that the model can be used for a large variety of system configurations, it should be possible to implement at least the following system characteristics:

- Upstream system
- Adjustable workload division across AS/RS
- Multi-deep storage racks
  - Storage strategy
- Multiple cranes
  - Collision avoidance strategy
  - Job scheduling and job division across cranes
- Multiple inputs and outputs
- Crane dwell points
- Output to directly unload goods
- Possibility to return empty pallets
- Orders with a specified output sequence
- Downstream system

5.2. Modelling method

As explained in chapter 2, AS/RS can be modelled in various ways. Because of the complexity of the system under study, using an analytical model is not an option. For example, keeping track of goods already in production once the downtime starts, or modelling the influence of specific downtime scenarios in combination with the behaviour of upstream and downstream systems would not be possible. Therefore, the choice was made to develop a simulation model.

The type of simulation model to develop and the platform in which to develop it depends on the characteristics of the system and external factors, like software accessibility and previous experience of the modeller. Because of the nature of the system under study, it was decided to make the model a DES model. The processes in the system can be described as a sequence of discrete events very well, and apart from the cranes, there are not that many separate agents in the system which rules out the need for an ABM.

To achieve the research goal of developing a modelling framework that can be reused by other researchers, the software to be used needs to be accessible to everyone. This resulted in the choice of using the open-source Salabim DES library in Python. The advantage of using Salabim over the more frequently used DES library SimPy in Python is the ability to use the Simula activate/passivate/hold paradigm (van der Ham, 2018). In addition to that, it includes extra features such as queues, tracing and monitors for data collection and presentation. Using this library ensures that any modeller can use the model, and with some coding experience, the model can be easily adjusted.

5.3. Input data

The input data determines the behaviour of the upstream and downstream systems and therefore also determines the behaviour of the AS/RS. The input data should specify the release of orders to the production systems, the contents of orders, their output sequence and the retrieval time of orders. For each order, the data as explained in table 5.1 is stored.

Entry	Explanation
Order ID	Unique ID of this order
Release time	Earliest time from when production of this order can start
Departure time	Time at which this order is scheduled to depart
Loading sequence	Defines number of pallets in this order and loading order

Table 5.1: Input data for each order

The format chosen is a list of orders which are sorted by their scheduled departure time. One order can contain multiple clients for which the goods are grouped. To account for the warm-up period of the simulation, data from previous orders before the day that is desired to be studied should be included to ensure the rack does not start empty.

5.4. Model structure

The simulation model consists of several components. These can be either data components or components with processes. The unique property of this model compared to other models, of which few exist, is that it takes the upstream and downstream processes into account, multiple AS/RS working in parallel can be studied, and downtime can be considered. The components and their interaction will be explained in the following subsections. An overview of the interaction of components can be found in figure 5.1.

5.4.1. Pallet

This component is a data component. A pallet has the attributes shown in table 5.2. Some of these attributes are determined at the creation of a pallet and some are changed throughout the simulation.

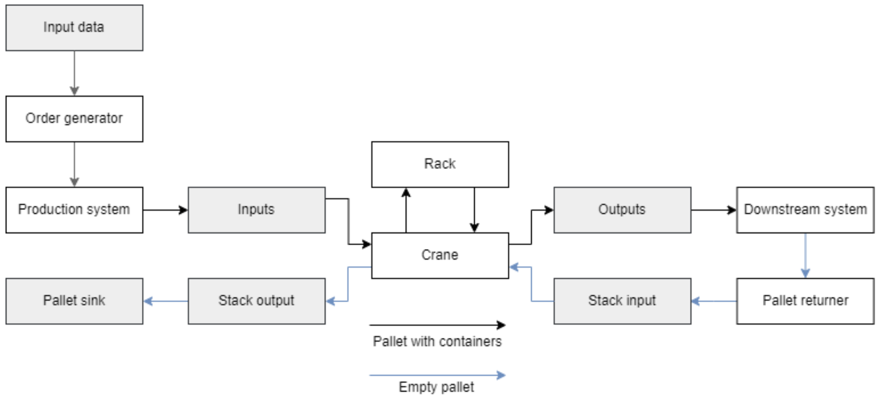


Figure 5.1: Model components and interaction

Attribute	Explanation
ID	Unique pallet ID
Order	Order number to which the pallet belongs
Sequence	Number of its loading sequence within order
Rack number	Number of the rack to which it belongs
x1	X coordinate where in the rack the pallet is now
y1	Y coordinate where in the rack the pallet is now
z1	Z coordinate where in the rack the pallet is now
x2	X coordinate where in the rack the pallet needs to go next
y2	Y coordinate where in the rack the pallet needs to go next
z2	Z coordinate where in the rack the pallet needs to go next
Departure time	Departure time of the order to which the pallet belongs
Output points	Set of possible output points where the pallet could go to
Input point	Input point in the rack of the pallet
Release time	Time at which the pallet is released to be moved
Status	Current status of pallet, shows where in the system it is currently

Table 5.2: Pallet attributes in model

5.4.2. Rack

This component is also a data component. The most important attributes of the rack are shown in table 5.3, more attributes exist but are mainly used for data collection. Some of these attributes are characteristics of the rack which are set at its creation, while others are changed throughout the simulation. The x-coordinates run along the width of the rack, the y-coordinates run along the height of the rack and the z-coordinates run along the depth of the rack.

Attribute	Explanation
Number	Number of this rack in the system
Type	Type of rack, supports multiple rack configurations
Free locations	Locations in the rack which are free
Occupied locations	Locations in the rack which currently store a pallet
Blocked locations	Locations in the rack which are permanently blocked
Number of cranes	Number of cranes servicing the rack, can be 1 or 2
Input points	Locations where the input points are located, can be multiple
Output points	Locations where the output points are located, can be multiple
Pallet stack input point	Location where stacks of empty pallets can be picked up by crane
Pallet stack output point	Location where stacks of empty pallets can be dropped off by crane
Failure lane	Location where pallets can be unloaded in case of failures in the system
Contents	List of pallets currently in the rack

Table 5.3: Rack attributes in model

5.4.3. Order generator

The order generator component translates the input dataset into orders and pallets in the simulation environment. The pseudo-code for this component can be found in appendix B.2. The key responsibilities of this component are as follows:

- Releasing orders at the time specified in the input data
- Ensuring production is started at the specified production start time and stopped the evening before to guarantee lead times and minimise production stops
- Determining the properties of each pallet, generating them and sending them to the production system
- Limiting the maximum amount of orders in production concurrently to the specified limit
- Dividing the orders over all AS/RS
- Determining the outputs for each order
- Shuffling the order of pallets within the production queue to ensure a randomised production order

To ensure that production stops the evening before, orders are released up to a specified departure time after which order release is stopped and started again at the specified starting time the next day.

When dividing the orders over the racks, the rack capacity is taken into account. For example, when the rack capacity is set to 50%, the order generator makes sure that the rack only receives 1/7 of the orders if there are 3 other racks with capacity 100% which all receive 2/7 of the orders.

When selecting the outputs for the orders, the model takes the output combination where the last order appointed to it leaves the earliest. By doing this, it is ensured the margin between the departure times is the largest possible.

5.4.4. Production system

The production system component takes the pallets that were placed in the production queue by the order generator and produces them and thus represents the upstream process of the AS/RS. When a pallet is produced, it is sent to the central conveyor and input points of the racks. The pseudo-code for this component can be found in appendix B.3. The key responsibilities of this component are as follows:

- Producing pallets which are placed in the production queue by the order generator
- Determine which order to produce a pallet from next based on a stochastic distribution
- Check if cranes are down and exercise policies by changing pallet rack and lane destination
- Determine the time it takes to produce a pallet based on a normal distribution
- Send the pallet to the conveyor and input points
- (Re)activating order generator and cranes
- Replanning orders to another rack in case of full downtime

A limit on the number of orders in production concurrently can be specified. In industry, this is done to maximise the efficiency of the picking systems. Pallets from orders are not produced exactly in the order of departure time, they are somewhat randomised. There is a preference for producing a pallet for an order with an earlier departure time, but it can occur that sometimes a pallet of an order that has to depart later is produced. The odds of a pallet from an order being produced can be seen in figure 5.2. This distribution is based on the equation: $f(x) = cos(x)^2$ on the interval $0 \leq x \leq 1$ after which it is normalised.

This distribution is scaled with the size of the production pool, so with fewer orders currently in production than the limit, because there are no more orders released for example, the shape of the distribution remains the same. Next to the order from which a pallet is produced, the sequence of pallets being produced within an order is also random. This was already made sure by the order generator. In case the orders of the system under study are produced in the exact sequence of departure time, the production pool size can be set to 1.

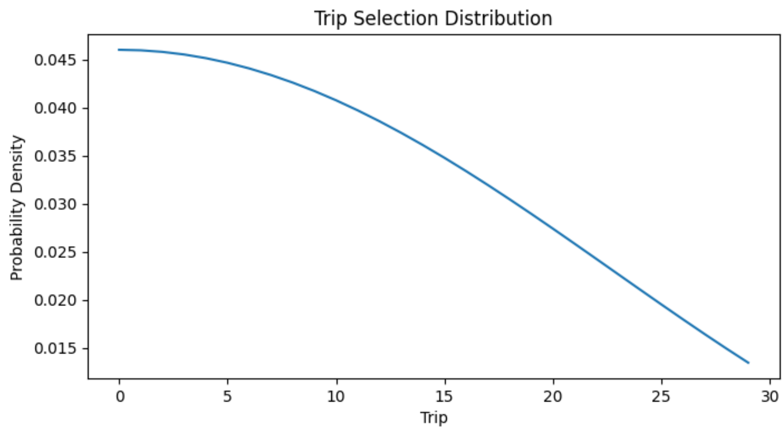


Figure 5.2: Probability density for selection of an order to produce from the production pool

The time it takes to produce a pallet is sampled from a normal distribution with a mean and standard deviation that can be specified in the input parameters.

When a pallet needs to be produced for a rack from which both cranes are down, none of the pallets for this order are already at the outputs and not too many pallets are already in the rack, it can be rerouted to another AS/RS. Because there are already more orders scheduled for that other AS/RS, it has to be decided as to which output to schedule the order for. This is done by comparing all orders scheduled for those outputs and determining where to schedule this one. This could be between other scheduled orders, or after them, depending on the departure times while ensuring the largest minimum margin between departure times at the outputs. If there are already too many pallets of this order in the broken AS/RS, it does not make sense to reroute it since then half of the order would be in one AS/RS, and the other half in another one. When just a few pallets are already in the broken AS/RS they can be unloaded from that AS/RS via the failure lane afterwards and manually moved to the rest of the pallets at their new AS/RS.

5.4.5. Crane

The crane component is the most complicated component. The pseudo-code for this component can be found in appendix B.4. The key responsibilities of this component are as follows:

- Keeping track of downtime
- Determining the task sequencing
- Determining if tasks are possible
- Determining the rack location of pallets
- Avoiding collisions with the other crane
- Moving pallets between inputs, the rack and outputs
- Activating the downstream system and pallet returners

The sequencing of tasks is determined with an objective function. For all tasks in the crane queue, it is determined whether they are possible to perform and a score is given. This score is based on departure time, loading order within an order, input point population, empty pallet stack priority and possible waiting time to avoid collisions. The base formula chosen is: $score = t_{current} - t_{departure} - sequence$ which ensures that the pallets with earlier departure times and which come earlier in the loading sequence are treated first. If this pallet is at the input point and there are 2 pallets at the input, the score is equalled to -1800, which ensures that only pallets that are already late for the output go first. When there are more than 2 pallets at the input, and thus blocking the central conveyor, the score is equalled to 7200, which ensures that only pallets that are 2 hours too late will go first. If the task concerns a stack of empty pallets, the score is equalled to 7200 as well, since there is no buffer capacity for empty pallets to wait. In case performing the task will cause waiting time to avoid a collision, a value of 200000 is subtracted from the score which ensures that it is unlikely that the task gets selected unless there are no other possible tasks.

The rack location is determined based on the loading sequence of the order and the output where the pallet is supposed to go. Pallets from the same client within an order can be stored behind each other or in front of clients that need to be retrieved later. In general, pallets will be placed in the column above their output when possible, otherwise, the overall travel distance from input to rack location to output is minimised.

To avoid collisions between both cranes operating on the same rail, for each crane, the planned trajectory while performing a task is stored. Then, while choosing a new task to perform, the planned trajectory is compared with the current trajectory throughout time and it is determined if the cranes would collide or not.

For the calculation of crane moving time, several formulae are used. In case the maximum crane speed is not reached, as depicted in figure 5.3, equation 5.1 is used. In case the maximum crane speed is reached, as depicted in figure 5.3, equation 5.2 is used. To determine whether the maximum speed is reached over a certain distance, equation 5.3 is used.

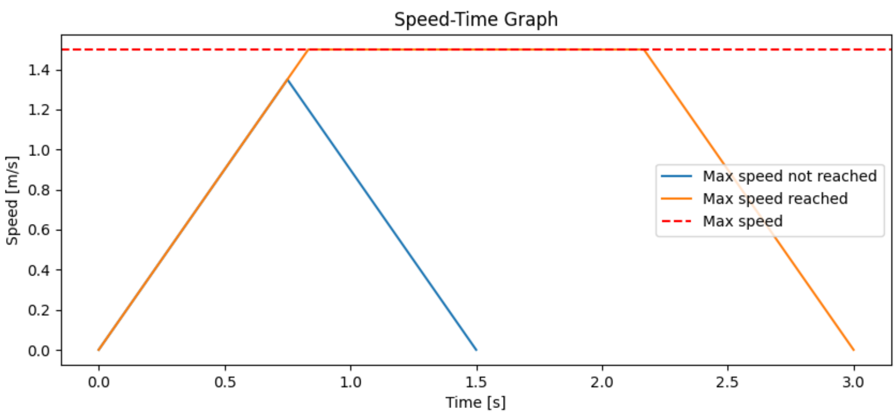


Figure 5.3: Crane speed over time when maximum speed is or is not reached

$$t = 2 * (\frac{s}{a})^{0.5} + t_{d,base} + (|z_1| + |z_2|) * t_{d,added} \tag{5.1}$$

$$t = 2 * \frac{v_{max}}{a} + \frac{s - \frac{v_{max}^2}{a}}{v_{max}} + t_{d,base} + (|z_1| + |z_2|) * t_{d,added} \tag{5.2}$$

$$s \leq \frac{v_{max}^2}{a} \tag{5.3}$$

Where:

t = travel time $[s]$

s = distance to travel $[m]$

a = crane acceleration $[m/s^2]$

$t_{d,base}$ = base time to pick up pallet $[s]$

z_1, z_2 = depth for pallet pickup and placement $[racklocations]$

$t_{d,added}$ = added time to pick up pallet per depth location $[s/racklocation]$

v_{max} = maximum crane speed $[m/s]$

5.4.6. Downstream system

The downstream system component can take the goods from the pallets and further processes them. In case the goods continue to the downstream system on pallets, the pallet returner component can be deactivated. Pseudo code for this component can be found in appendix B.5. The key responsibilities of this component are as follows:

- Determining which of the outputs to process goods from next
- Determining when to start processing goods of an order
- Determining how long it takes to process the goods
- Removing pallets from the outputs and moving them to the empty pallet queue
- Activating cranes and pallet returner

Because in reality, the downstream process might not always start processing goods exactly at the specified departure time, the time of an order when it starts to be processed is sampled from the gamma distribution shown in figure 5.4. As can be seen, sometimes the order starts being processed early but it happens more often that an order starts being processed too late with possible outliers which are more than an hour late. The mean start time is half an hour before the departure time of an order. This mechanism will always make sure some orders are too late, regardless of the used policy.

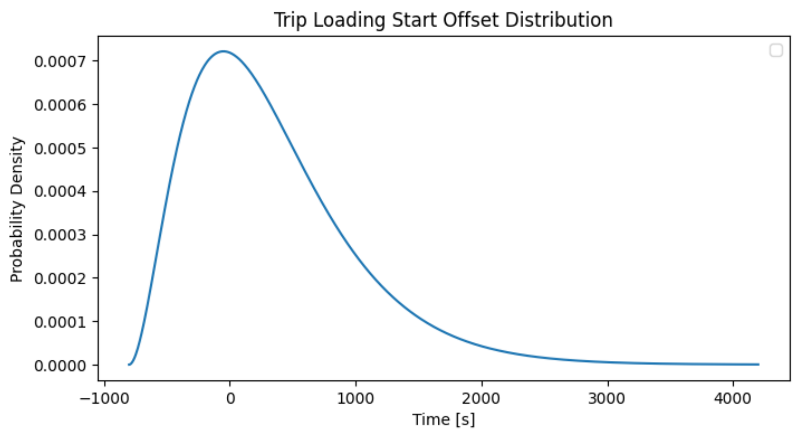


Figure 5.4: Time in seconds difference orders start to be processed compared to the scheduled start time

The time it takes to process one pallet is also a stochastic process. The mean and standard deviation of the normal distribution used to model this can be specified in the input parameters.

5.4.7. Pallet returner

The pallet returner component collects empty pallets which come from the outputs and sends stacks of empty pallets to the crane. Pseudo code for this component can be found in appendix B.6. The key responsibilities of this component are as follows:

- Collecting empty pallets
- Waiting until there are enough pallets to form an empty pallet stack
- Sending empty pallet stacks to their input point and crane

In the case of multiple cranes, when an empty stack is complete and both cranes are passive, it is sent to the closest crane. When just one crane is passive, it is sent to the passive crane. If both cranes are active, it is sent to both crane queues and when one of the cranes picks up the tasks to move the stack the task is also removed from the other crane queue.

5.5. Assumptions and simplifications

The main assumptions of the model are as follows:

- Orders are spread out equally over the AS/RS and outputs
- New pallets for orders from which too many pallets are in a broken AS/RS, or from which at least one of the pallets is at an output, are directly unloaded via the failure lane of the nearest AS/RS
- Production is started at the specified starting time and ends after enough is produced for the next day to ensure acceptable lead times and minimise production stops
- Within those times, production starts as early as possible after order release
- The next pallet from an order to be produced is completely random and does not depend on the loading sequence of the order
- The next order from the production pool to produce a pallet for is random with a slight preference for earlier departure times
- Rack and output to go to for an order are determined at the moment an order is released and starts production
- Every crane only processes pallets from its own input point
- The failure lane is always available and never full of pallets
- If one of two cranes is down, the remaining crane will service its full service area instead of just half of the rack
- If one crane breaks down and the decision is made to continue operation with the other crane, it needs to be moved to the side
- Pallets that arrive at an AS/RS while a crane is being moved to or from the repair position are sent to the failure lane of the nearest AS/RS

5.6. Options

Both experiments studied in this research can be simulated with this model; one crane or both cranes from an AS/RS being down. To alternate between these experiments, just one parameter needs to be changed.

Also, the model contains an option to determine when production for an order can start. This can either be as early as possible after the release of an order, or it can be an offset from the departure time of the order. Similarly, the moment when pallets of an order can start moving to their outputs is determined with an option. This can either be as early as possible, while still taking the correct loading sequence into account, or this can also be an offset from the departure time of the order.

Lastly, the simulation can be run in two modes. The first mode is meant for a single simulation run. In this mode, more data is collected as described in the next section. The second mode is meant for multiple simulation runs with varying random seeds. In this mode, just the desired core KPIs are collected and stored, making the simulation faster. To reduce the run-time of the model, several options were explored. First of all, profiling was used to identify the most computationally intensive parts of the code. With this tool, these parts of code were reworked into code that is more efficient with the computational resources.

Because in a Discrete Event Simulation, the same functions are called repeatedly, it was investigated if using Just-In-Time (JIT) compilation would speed up the simulation. With JIT compilation, the first time a function is called, it takes longer to compile, but after that, the functions can be executed quicker because they have been translated to machine code. To achieve this, the PyPy interpreter was used instead of the standard CPython interpreter. This, unfortunately, did not lead to efficiency gains. After a discussion with the creator of the Salabim library, it was concluded this is because the newer versions of Python are already quite efficient.

When inspecting the processor usage of the model, it was noticed that the model does not use the full capacity of the processor. To make use of the full capacity, a script was written that initiates multiple instances of the model in parallel making it possible to run them simultaneously. After testing, it was found that running 8 simulations in parallel resulted in the shortest overall simulation time on the computer used. The script ensures each simulation is run with a different random seed, collects the extracted KPIs and processes them.

All efforts combined led to a reduction in simulation time from ±100 seconds to ±10 seconds per simulation run for the heaviest dataset.

Next to these built-in options, there are also single settings that can be adjusted to the desired configuration. An overview of the possible and used simulation settings can be found in appendix C. Additionally, if necessary, any modeller can easily adjust the Python code of the model to in- or exclude specific system characteristics.

5.7. Output

The goal of running the simulation is to be able to see what happens in the system in varying scenarios under different policies. When run in the first mode, the model produces an Excel file containing the following data:

- Log with performed actions of every component to be able to trace events
- KPIs specified in chapter 4
- Plots of KPIs of each crane throughout time
  - Task queue length
  - Utilisation
  - Cycle times
- Plots of KPIs of each rack throughout time
  - Overall outputs fill grade
  - Individual output fill grades
  - Rack fill grades

- Order delays
- Input queues length

- Plots of the overall simulation
  - Overall order delays
  - Overall outputs fill grade
  - Overall racks fill grade
  - Number of pallets on central conveyor
  - Number of pallets produced, placed in rack, at outputs and departed over time
  - Pallets produced and processed per hour

In conclusion, the answer to sub-question 3 is that parallel AS/RS should be modelled with a Discrete Event Simulation model to be able to take enough details into account to study partial downtime. The model should consist of components modelled as classes of which multiple instances can be used in a simulation. The necessary data components are pallets and racks. Next to that, there should be a component representing an order generator, an upstream production system, a crane and a downstream system. Additionally, the model should contain several options which increases the flexibility of the model to be used to study a large variety of scenarios.

5.8. Model reusability

To answer sub-question 4, the developed model can be reused by other AS/RS users to study a large variety of system configurations. Because this model was developed in Python, with all open-source libraries, anyone can use and adjust this model to study AS/RS. The model was parameterised as much as possible within the timespan of this research leading to easy adjustment of most parameters such as the input data, the upstream production system settings, the number of AS/RS, the number of cranes, the rack characteristics, the crane characteristics and the downstream loading process settings.

As a result of this, AS/RS can be researched in isolation, in parallel and with some adjustments to the model even in series. With the integration of the upstream and downstream systems, their influence on the AS/RS performance can be investigated and vice versa. Other examples of topics that could be researched with this model after some adjustments are as follows:

- Energy consumption of AS/RS under varying control strategies
- Dwell point optimisation
- Optimising efficiency of dual-crane AS/RS by smart job scheduling
- Optimising SKU storage location policies in multi-deep AS/RS to prevent efficiency losses due to relocations
- Optimising production and loading schedules

For a large variety of systems and scenarios, this model can be used with few adjustments, and generally, it can be used as a base for any AS/RS model. This can help other researchers since there do not appear to be many publicly available AS/RS Discrete Event Simulation models which are free to use and written in Python, which is a programming language that is easier to understand and learn for non-native programmers. A manual on how to use this model can be found in appendix E and the model itself with all corresponding files can be found at https://github.com/lucvdbrink/Parallel-ASRS-DES-Model.

6 Case Study

This chapter will describe the system at Jumbo that will be used as a case study and its implementation in the simulation model, thereby contributing to the answer to sub-question 5.

The system under study is the OCB of the newly built, highly automated CDC for the fresh assortment in Nieuwegein for Jumbo Supermarkets. The warehouse has been designed and built by the German company Witron, which is specialised in automating logistics. This OCB consists of four parallel dual-crane AS/RS resulting in the possible occurrence of partial downtime, which makes it a suitable case study for this thesis. Jumbo Supermarkets is curious about the effects of partial downtime on their system and possible policies to mitigate the consequences of this.

6.1. Delivery Schedule

The DC services the supermarkets of Jumbo. These supermarkets place orders for products every day. The orders of supermarkets, also called the clients of the DC, are combined into trips. There is a fixed delivery schedule based on expectations for orders and this schedule can be adjusted to larger or smaller orders from clients. Trips are spread throughout the day to spread the workload for the DC. Trips can be for just one client or for multiple clients. If a trip consists of multiple clients, it is important that the rollcages (RC), which are filled with their ordered products, are loaded into the trailer in the correct order so that when the truck arrives at the client, all RC for that client are directly accessible in the trailer. The definitive orders of clients arrive at fixed moments spread out over the day. The relevant data for each trip includes the earliest production start time, depending on order processing, departure time, quantities of containers per client, and a specified loading sequence for the containers into the truck.

This weekly trip schedule from Jumbo was converted into a format that can be loaded into the simulation model. The dataset obtained from Jumbo was in a different format and contained some mistakes, therefore it had to be treated. The following steps were taken to go from the dataset of Jumbo to the datasets used for this research:

First of all, the dataset from Jumbo contained all trips for a whole week, and for this study, it is desired to simulate one full production day. The workload for each day of the week for supermarkets is not equally divided. For example, towards the weekend the workload is considerably higher, while on Sundays it is considerably lower. To test the policies under different workloads of the system, three different input datasets were created of which the first two could be directly extracted from the obtained dataset. The first dataset represents a quiet to average day in an average week. This dataset is based on the Monday from the schedule of Jumbo. The second dataset represents a peak day in an average week. This dataset is based on the Friday from the schedule of Jumbo. The division of pallets and trips for each day in the week can be seen in figure 6.1

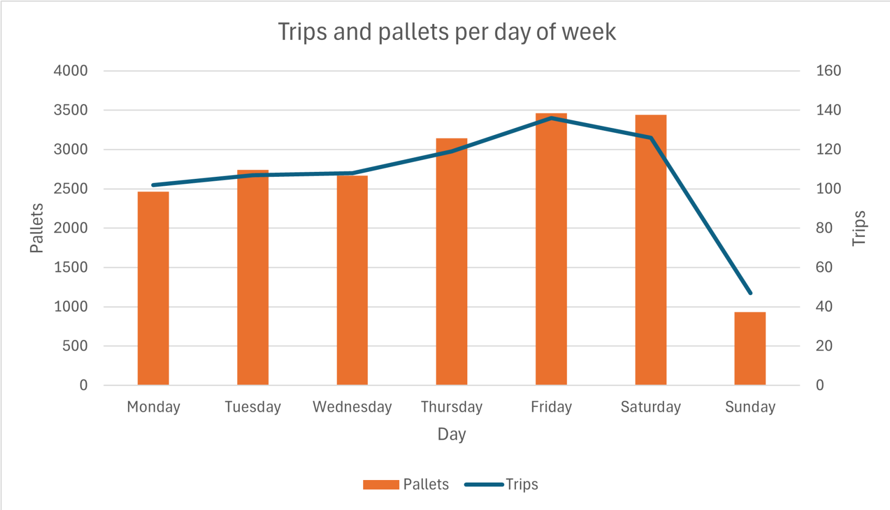


Figure 6.1: Overview of pallets and trips processed for each day in the week

To represent a realistic full day, the operations concerning the day before and after the targeted day which affect the operations on the day itself should also be considered. Therefore, while selecting the data, it was determined what the earliest release time for a trip on the target day was. For example, for Monday, the release time of the first trip for production was at 09:45 on the day before, Sunday. All trips departing on Sunday after this time are occupying output points and rack locations and therefore influencing the placement of pallets produced for Monday in the rack locations and output points. For this reason, all trips with a departure time on Sunday after the earliest production release for Monday were taken into account for the input data.

Similarly, trips departing the day after the targeted day also influence the system on the targeted day itself, because they could already start production on that day. Therefore, all trips with a production release time on the targeted day were taken into account for the input data. So overall, the data selected to study the operations on one full production day starts with trips with a departure time after the earliest release of a Monday trip and extends to the last trip released for production on the targeted day. The same selection was made for the Friday.

The obtained dataset was a large list of entries of all stores which includes their order moment, which results in the release time for production for their order, the departure moment of the trip headed to that store, the number of scheduled rollcages for that store and the trip ID this store belongs to. First, the list was sorted by the departure moment of the trips. Then, a Python script was written to convert the data to the desired format. This script combines the stores from the same trip into one data entry for each trip. The order amount for each store was stated in the number of rollcages, but the simulation model works with pallets. Two rollcages are placed onto one pallet and mixed stores can be on the same pallet in case of uneven order quantities. To deal with this, the script converted the amounts of rollcages to amounts of pallets and directly specified the loading order of those pallets. Pseudo code for this script can be found in appendix B.1.

Because the obtained dataset was preliminary, the number of rollcages per trip sometimes exceeded the maximum a trailer could carry. This maximum is 54 rollcages, which translates to a maximum of 27 pallets per trip. This number can be slightly higher since it sometimes occurs that rollcages can be consolidated, but it can not be much more. Part of the volume of trips with over 28 pallets assigned to them was moved to other trips to keep it realistic. These pallets were subtracted from those trips and added to trips with a similar departure time. If no trips with a similar departure moment and space for these pallets existed, multiple surpluses were combined into a new trip with a similar departure time.

Also, for some trips, their release time was earlier than other trips with an earlier departure time and a later release time. In the real system at Jumbo, this is not desired because this might cause hold-ups in the system. Under normal circumstances, the workload of the system is high enough so that these later trips which are released earlier do not start their production too early, and if they do, the problem is solved operationally. To reflect this in the model, these earlier release times were delayed in such a way that all release times of trips were in chronological order, similar to the departure times. The spread of departure time throughout the day for each dataset can be seen in figure 6.2, an overview of the release times of trips and their departure times can be seen in figure 6.3 and representations of order lead times are shown in figure 6.4 which are relatively similar for each day.

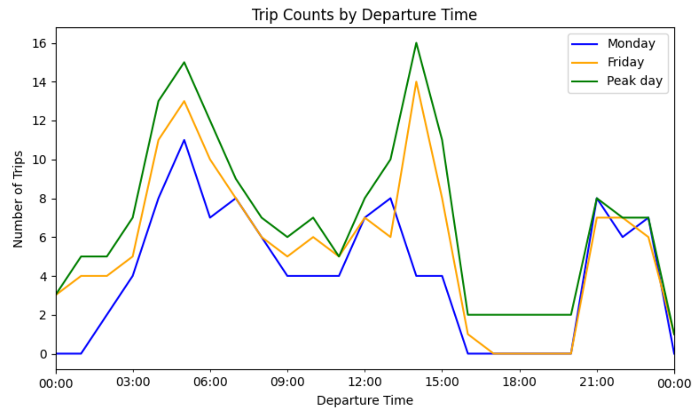


Figure 6.2: Spread of departure times throughout the day for each dataset

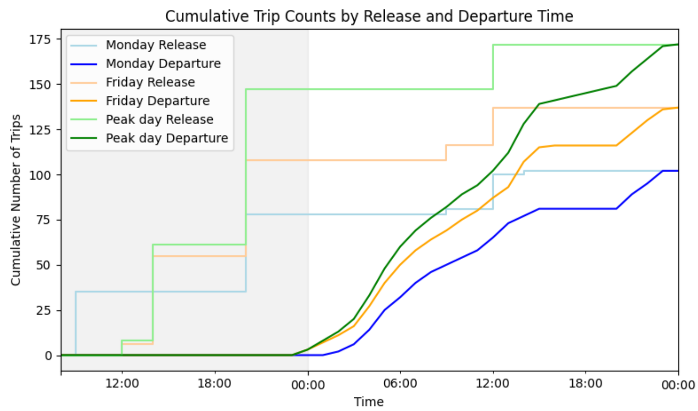


Figure 6.3: Spread of release times of trips throughout the day for each dataset

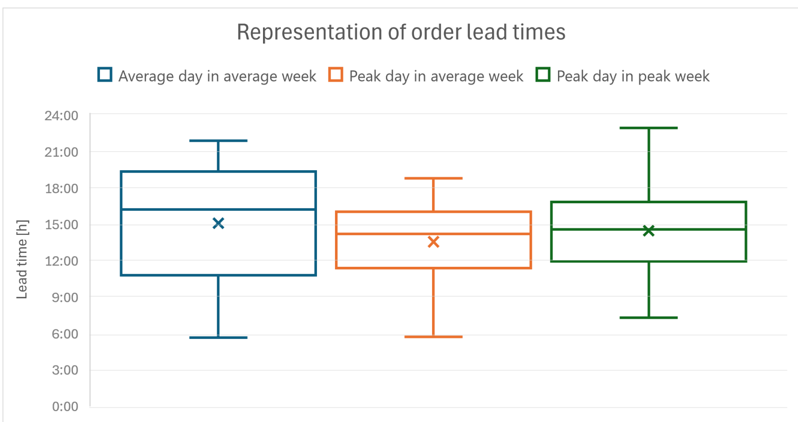


Figure 6.4: Representation of order lead times

On the target day in the first dataset, 2476 pallets are shipped in total. On the target day in the second dataset, 3449 pallets are shipped in total. On the target day in the third dataset, 4476 pallets are shipped in total. This last input dataset which could not be directly based on the obtained data represents a peak day in a peak week and was created based on the second dataset.

The number of containers and thus the amount of pallets processed in a day scales with the volume handled that day. According to Jumbo data, on a peak day in an average week, a volume of 3889031 L is shipped. On a peak day in a peak week, this is 5047072 L. To determine the number of pallets processed on a peak day in a peak week, the number of pallets processed on a peak day in an average week is multiplied by this ratio: $3449 * (5047072/3889031) = 4476$ pallets. When converting this to colli/container it results in ±76 colli/container which is deemed realistic by Jumbo. To reach this volume, extra trips were added to the dataset for a peak day in an average week in a way that extra trips would also be added in reality. In figure 6.2 it can be seen that the spread of departure times throughout the day is similar to the other days but with wider peaks.

6.2. Warehouse Flow

6.2.1. Production system

A general representation of the warehouse can be seen in figure 6.5. Incoming pallets with products are stored in an AS/RS. When the products are needed, the pallet is retrieved and the products are depalletised and placed onto trays after which they are stored in another AR/RS to make sure that the products can be sorted in the desired order. This is based on the order release and predetermined production schedule that follows a push policy. Therefore, once an order is released, it is difficult to cancel production for that order. These products are then fed into the picking system, which consists of multiple subsystems. RC are not filled exactly in the order in which the trips have to depart. This is because the system optimises the efficiency of the picking systems, and therefore sometimes chooses to fill RC of later trips or RC later in the loading sequence first. Normally, about 30 trips are in production at the same time. It is desired that production for a day happens in one continuous block instead of spread out over the day. Production starts at 06:00 and the day before, orders are produced in advance in such a way that lead times between production being ready and departure times of the trips are acceptable while ensuring that production is consolidated in one continuous block of production as much as possible. After the RC have been filled, they are placed onto pallets in pairs of two.

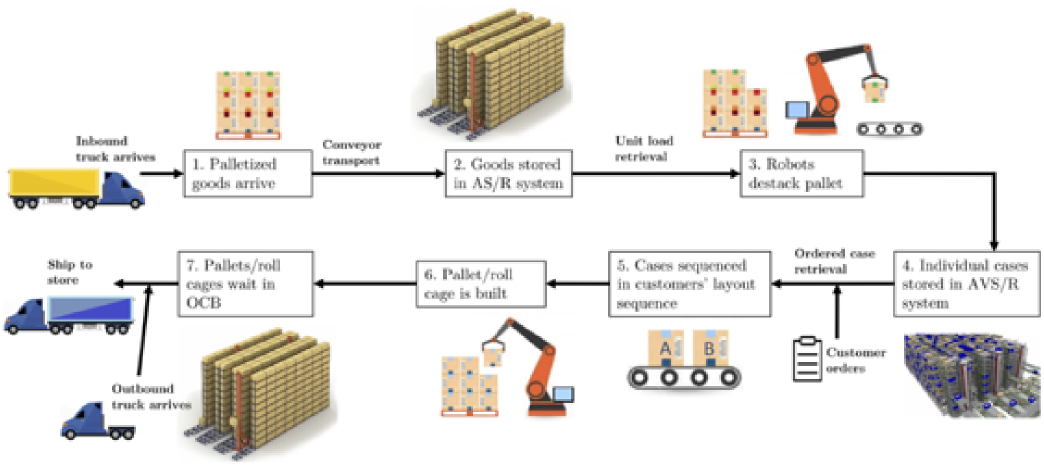


Figure 6.5: Total warehouse system (Azadeh et al., 2019)

The time it takes to produce a pallet is sampled from a normal distribution with a mean of 18.09s and a standard deviation of 2 as can be seen in figure 6.6. The maximum production speed was determined based on the capacities of the subsystems of production. In the OPM, 11940 colli can be processed per hour. Containers coming from this subsystem contain 57.5 colli/container on average leading to 208 containers/h. In AIO-picking, 12250 colli can be processed per hour. Containers coming from this subsystem contain 123.2 colli/container on average leading to 100 containers/h. In AIO-FT, 6144 colli can be processed per hour. Containers coming from this subsystem also contain 123.2 colli/container on average leading to 50 containers/h. In the CPS subsystem, 1839 colli can be processed per hour. Containers from this subsystem contain 46.8 colli/container on average leading to 40 containers/h. When converting containers to pallets: 398/2 = 199 pallets/h.

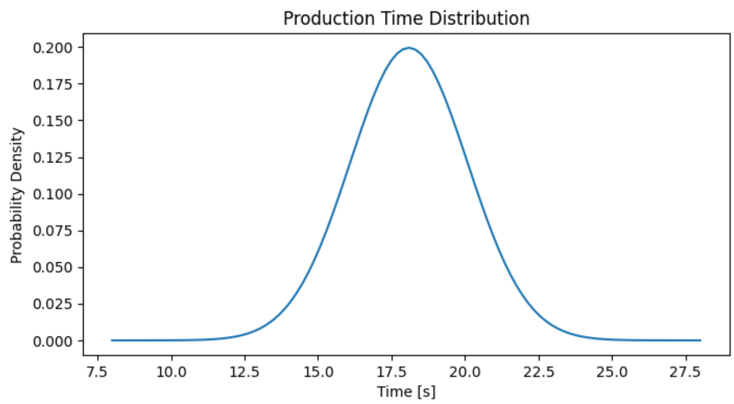


Figure 6.6: Production time distribution

6.2.2. OCB

After production, the pallets are placed onto a central conveyor which moves them to the input points of the right buffer. Three pallets can wait at these input points before blocking the central conveyor. The output points of the buffer are multiple gravity lanes which lead to loading docks. A gravity lane is a slightly tilted lane with rollers so that pallets can be placed on it on one side and will automatically roll to the other side from which they can be loaded into the trucks. A visualisation of this can be seen in figure 6.7. When the pallet at the input point is the next one in line to move to the output, and there is space in the gravity lane, it is moved there directly. Otherwise, the pallet is stored in the rack. The cranes continuously check if pallets in the system are ready to be moved to the gravity lane.

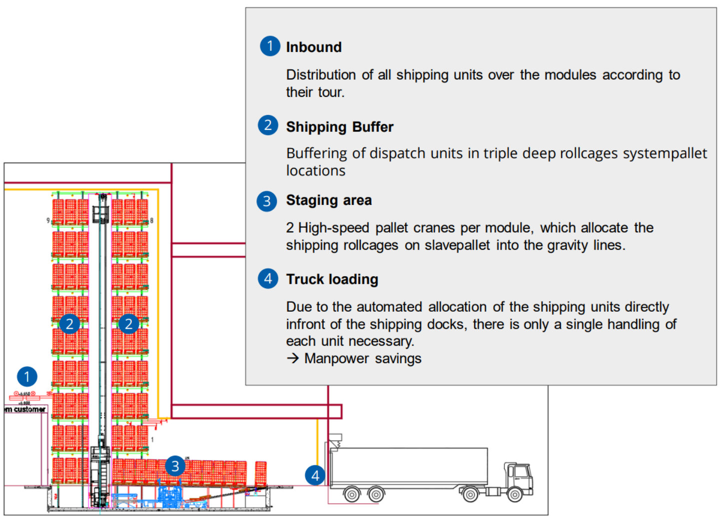


Figure 6.7: Buffer side view

The buffer consists of four separate racks. The central conveyor connects the inputs, but the outputs are disconnected. Once a pallet is moved into one rack, it cannot be moved to another rack anymore. The racks are 8 locations high on one side, and 7 locations high on the other side, this is because the bottom row on that side consists of the output points. The racks are 16 locations wide and triple-deep which causes the need for smart pallet placement together with the specified loading order so that each pallet needed is accessible and not stored behind another pallet. Certain locations in the rack are permanently blocked due to input or output points being located there. Each rack has two cranes, and two input points. These cranes operate on the same rail, which makes it possible for them to collide, which means that both cranes have a limited service area that partly overlaps. Both cranes mainly service their own input point. There are 14 output points per buffer as can be seen in figure 6.8. These output points are the gravity lanes with space for 13 pallets each. There are 8 loading docks for trucks next to the gravity lanes and mostly, 2 gravity lanes are used for storing and loading one trip.

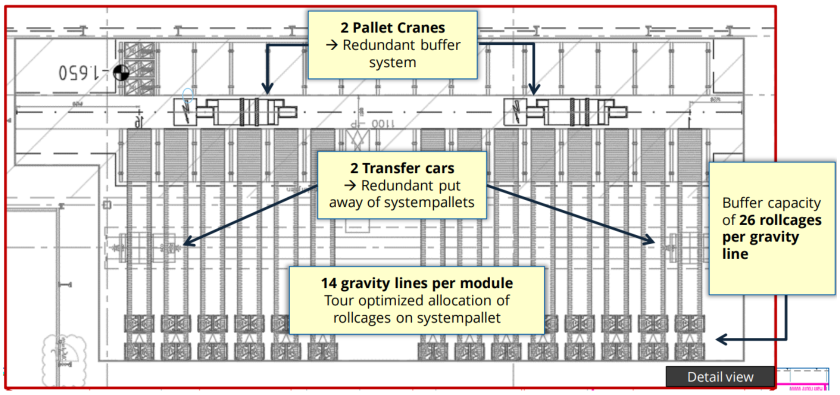


Figure 6.8: Buffer top view

6.2.3. Loading process

When a trip is ready to be loaded, the truck driver or transport employee scans the RC at the end of the gravity lane, takes them off the pallet and loads them into the trailer. The empty pallet is then returned to the system by a trolley where they are stacked into stacks of empty pallets. These stacks are then moved back to a crane which moves them back onto the central conveyor which takes them away.

It was determined that in total, it takes half an hour to load a trip of 26 pallets. This means it takes $1800/26 = 69.23$ seconds to load one pallet on average. The time it takes to load one pallet also depends on where in the loading sequence of the trip it is. The first pallets take longer to load since their containers have to be placed deeper into the trailer than the last containers. To accommodate for this, a factor depending on the loading order is introduced. The probability density distribution of loading a pallet can be seen in figure 6.9 and the factor depending on the loading order can be seen in figure 6.9. It was estimated the first pallet takes 10% longer than average and the last pallet takes 10% shorter than average. This means that loading the first pallet takes 76.15 seconds on average and loading the last pallet takes 62.31 seconds on average.

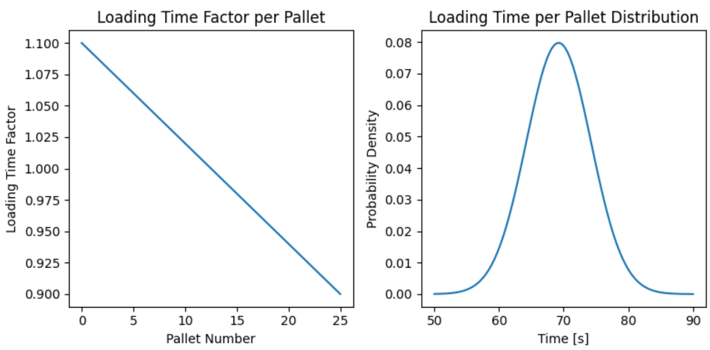


Figure 6.9: Loading time per pallet and loading time factor

6.3. Assumptions and simplifications

The following assumptions were made when implementing the system under study in the simulation model:

- Drivers start loading for their trip ±30 minutes before departure time
- It takes 30 minutes to load a trip of 26 pallets
- Production speed is set to ±199 pallets per hour
- The buffer is switched on 30 minutes after it is back up
- Each trip can be loaded from any dock connected to any buffer
- It takes 45 minutes to move a crane behind the maintenance fence, and 30 minutes to move it back into operation

The main simplifications of the system implementation are explained in table 6.1.

Model	**Reality**	**Explanation**
Trip outputs selected from 7 fixed pairs of 2 outputs	Secret algorithm	Created most likely algorithm based on hints of manufacturer
Pallets can go directly from production to inputs	Pallets move via a conveyor network to inputs	Model does keep track of amount of pallets on conveyor
Production speed of pallets is fixed	Production speed depends on several factors and can decrease	Slightly lower fixed production speed is chosen to compensate
Empty pallets can be directly stacked and returned	Pallets have to wait for a transfer car to move them to stacker	Transfer car waiting time will not influence AS/RS performance
Trips are released in chronological order of departure time	Some trips with a later departure time are released earlier	In real system workload is high enough so that these earlier released trips are not produced too early
Own crane task scheduling algorithm	Secret algorithm	Created most likely algorithm based on hints of manufacturer
Own rack location selection algorithm	Secret algorithm	Created most likely algorithm based on hints of manufacturer
Relocations not considered	Relocations could occur	Will not happen because of overcapacity

Table 6.1: Main simplifications of model

6.4. System Logic

6.4.1. Crane Task Selection

The cranes perform various tasks, from which multiple can be pending at the same time. For example, moving a pallet from the input to the rack or an output, moving a pallet from the rack to the output, or moving stacks with empty pallets. For the overall efficiency of the system, the crane must select the best task to perform next. This can depend on multiple factors. For example, the departure time of the pallet to be moved, the number of pallets at the input, which matters because the central conveyor should not be blocked, or the number of empty pallets at the stacking system, which matters because otherwise empty pallets cannot be returned anymore and the gravity lanes will get jammed. Also, because both cranes operate on the same rail, tasks should be selected in such a way that makes sure crane interference and waiting time is minimal. The system takes all of this into account and determines which task to perform next.

6.4.2. Rack Location Selection

Pallets arriving at the buffer which cannot be placed directly in the gravity lane have to be placed in the rack. The rack is triple-deep which makes it important to intelligently place the pallets so that the right ones are always accessible, while not wasting storage space or travel distance of the cranes. The system optimises this in such a way that the overall travel distance of the cranes is minimised, the right pallets are accessible and storage space is used optimally. An overview of the rack layouts can be seen in figure 6.10

6.4.3. Dock and Gravity Lane Selection

The system has 4 buffers with 8 loading docks each. It is desired to spread the load on the buffers and docks to improve efficiency, because the higher the fill grade of the system, the longer the cycle times of the cranes will be. This is because, with higher fill grades, pallets are stored higher up in the buffer causing more travel distance. Not having them load too quickly after each other on the same dock is also important to avoid waiting time caused by trucks being too early or too late.

6.4.4. Failure lane

Each buffer has one gravity lane which is a designated failure lane. This lane can be used to unload pallets from the buffer at all times. This could be done when pallets need to be checked or in case of rerouting of pallets. For example, when another buffer is down, the pallets destined for that buffer can be rerouted to other buffers. If action is taken in time and the dock destination of the pallets is changed, they can be kept in the other buffer, but when no action is taken, they are directly unloaded at the failure lane of this other buffer. When unloaded at the failure lane, the containers have to be stored somewhere before being loaded.

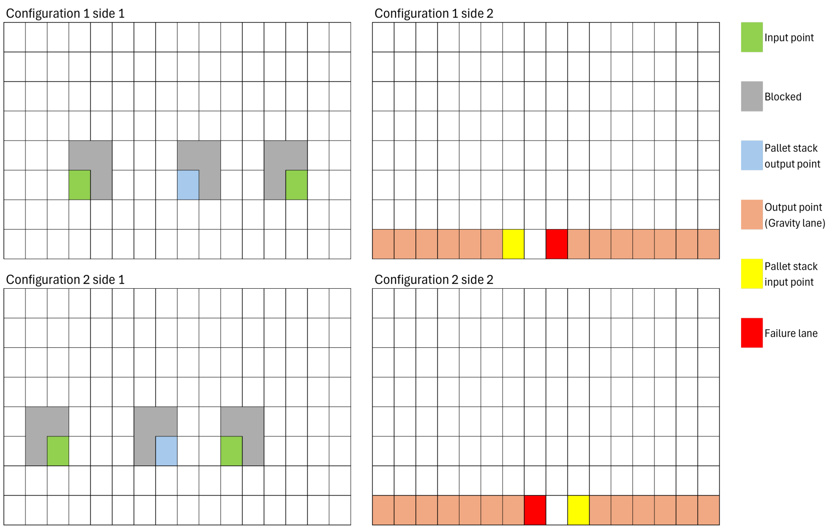


Figure 6.10: Rack Configuration

A visual representation of the layout of the buffers and locations where containers can be stored can be seen in figure 6.11. One option is to load them into cooled trailers which can be connected to the docks. In total, there is space to temporarily store 24 containers in location 1. A maximum of 324 containers can be stored in cooled trailers as depicted in location 2 which can be moved to any dock, however, this is when 6 trailers are used and when they are completely filled making it more difficult to retrieve the containers later. Ideally, the number of containers in the trailers should not exceed 152 containers. 148 containers can be stored in location 3 and 139 containers in location 4 leading to a total of 635 containers which can be stored temporarily around the buffers. It has to be noted that these capacities are in containers and the model works with pallets, two containers fit on one pallet. This means that a total of 318 pallets can be stored in these locations, with containers needing to be stored sub-optimally in the trailers above 232 pallets.

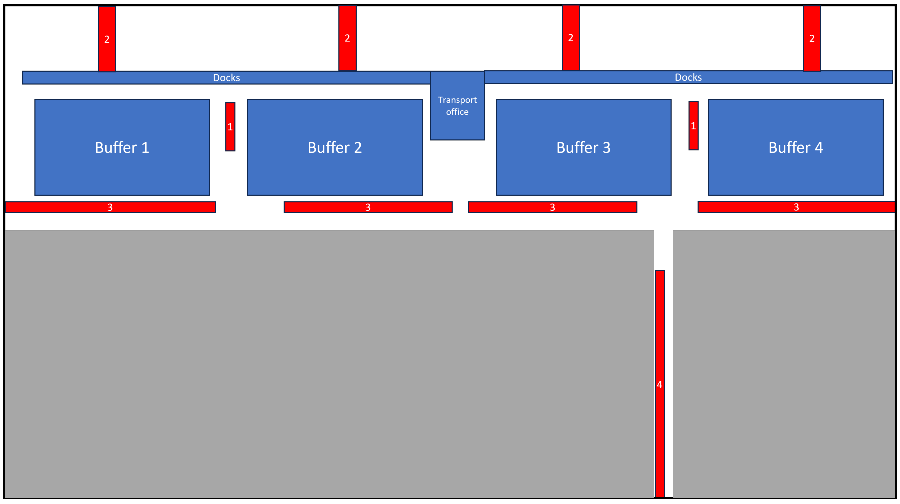


Figure 6.11: Visual representation of locations where containers can be temporarily stored

6.5. Performance

For Jumbo, the client is the number one priority. Therefore, it is important that the deliveries are on time and complete. According to them, the buffer performs well if the containers are in the gravity lanes on time. Also, the amount of containers to be unloaded via the failure lane and thus to be moved manually is desired to be minimised. The more containers have to be unloaded by hand and temporarily stored, the further away from the loading docks they have to be stored. This results in only more work leading to more costs. Next to that, the conveyor leading to the buffers cannot be too full because there is a risk the production would have to be stopped then.

6.6. Current policies

Currently, when one crane breaks down, the workload this buffer receives depends on the number of available outputs. With one crane behind the maintenance fence, 8 out of 13 outputs are available leading to a workload of 62% compared to the original workload, this is comparable to policy 2.

When a whole buffer breaks down, the inputs of this buffer are blocked. This causes the pallets that were already in production for this buffer to be unloaded through the failure lane of the nearest buffer. This corresponds to policy 1.

In conclusion, this chapter explained the system at Jumbo and its implementation into the simulation model, thereby contributing towards the answer to sub-question 5.

7

Verification & Validation

7.1. Verification

To ensure that the model is implemented correctly according to the intended specifications and design, the model and input data need to be verified. During the development of the model, verification was continuously taken into account by implementing one class at a time and using a top-down approach. A multitude of tests was performed both during the development of the model and afterwards of which several will be highlighted in this section.

The number of pallets processed in a day according to the generated dataset is 2476 for the first dataset, 3449 for the second dataset and 4476 for the third dataset. The number of roll cages processed on the first day is 4932 in total according to the original data which corresponds to 2476 pallets taking into account that some pallets have an empty roll cage. The number of roll cages processed on the second day is 6928 which corresponds to 3449 pallets. The number of roll cages processed in the third dataset cannot be verified with original data since it is artificial. The minor differences in the numbers of roll cages and pallets are because the number of roll cages is not always an integer number in the original data. It occurs often that this number is 4.1 for example which is rounded to 4 in the script that converts it to pallets. The differences in volumes are acceptable, however.

Balance checks were performed by using counters to compare the number of pallets produced, moved to the rack and loaded to the dataset. This process is partially automated, incorporating built-in checks that stop the simulation upon error detection. For instance, when not all produced pallets are loaded or when cranes get too close. Additionally, output counters and graphs of the model were monitored. As can be seen in figure 7.1, for that scenario, the number of pallets loaded is equal to the number of pallets produced at the end, there are never more pallets moved past the rack than produced, never more pallets past the outputs than past the rack and that were produced and there are never more pallets loaded than past the outputs, rack and that were produced.

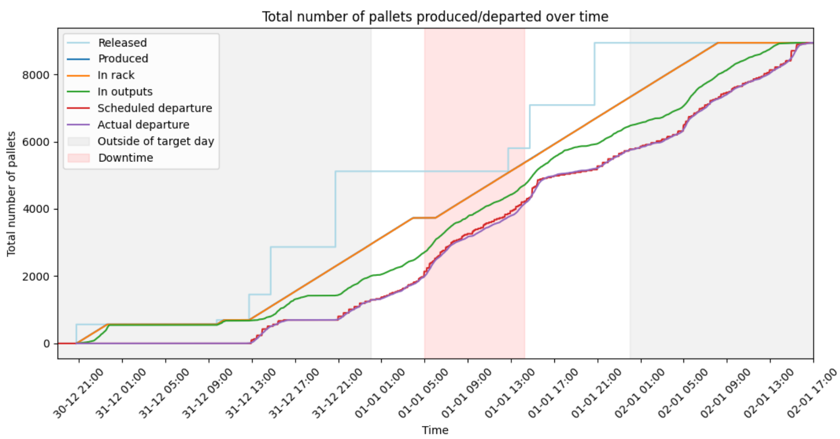


Figure 7.1: Number of pallets at different stages of the system throughout the simulation

The seed independence was checked by performing a set of simulations with varying random seeds. Although, naturally, the values of the KPIs varied with varying random seeds, no extraordinary variations in results were detected.

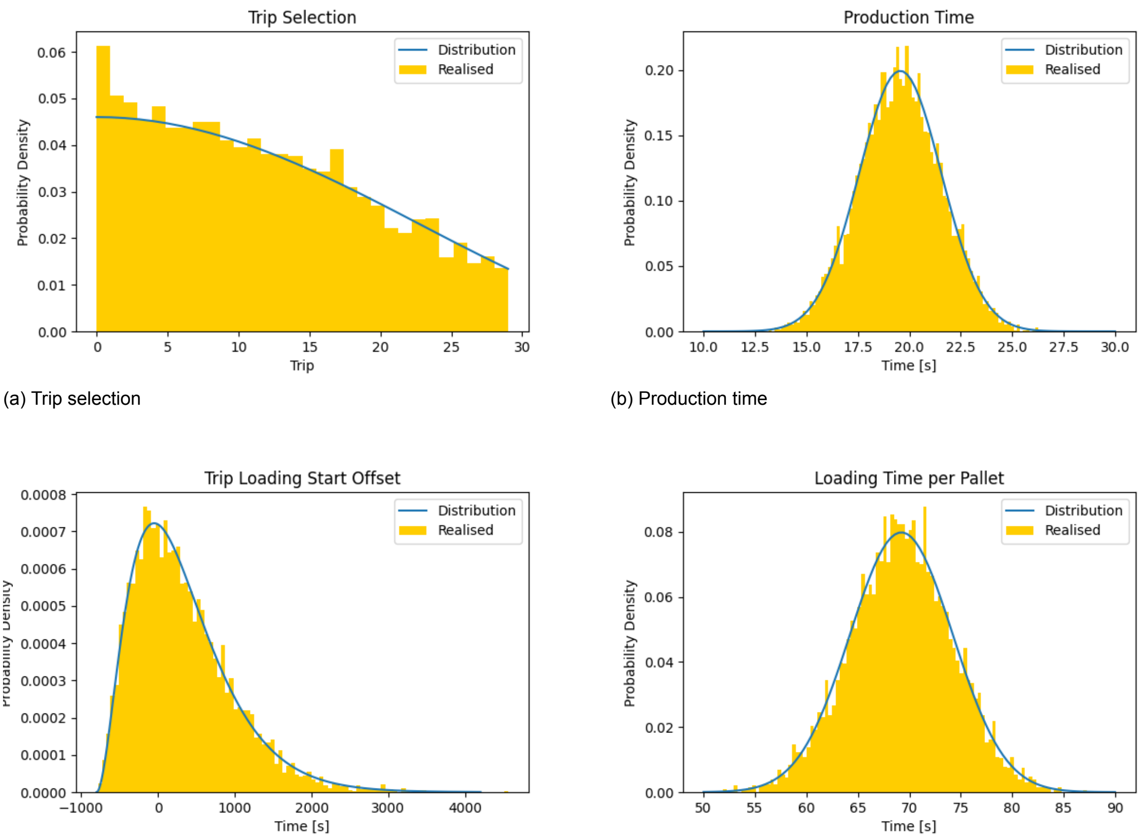
Consistency checks were performed while building the model and afterwards. These are examples of hypotheses that were tested:

- When the maximum crane speed is reduced, the average crane cycle time, average utilisation and longest continuous utilisation increase
  - Average cycle time: 33.39 s for v_{max} of 1.5 m/s, 36.69s for v_{max} of 1 m/s
  - Average utilisation: 41.63% for v_{max} of 1.5 m/s, 45.57% for v_{max} of 1 m/s
  - Longest continuous utilisation: 80.11 min for v_{max} of 1.5 m/s, 253 min for v_{max} of 1 m/s
- The number of tasks performed by cranes in a rack decreases for the rack with downtime and increases for the rack that takes the extra pallets depending on the policy
  - No downtime: 2195 tasks performed in rack 0, 2209 tasks performed in rack 1
  - Downtime in rack 1: 2609 tasks performed in rack 0, 1379 tasks performed in rack 1
- Average rack fill grade increases when output point capacity is decreased
  - 16.17% for gravity lane capacity of 13 pallets, 25.14% for gravity lane capacity of 8 pallets
- With a longer downtime duration, the aggregate trip loading delay increases
  - Total of 1153 min delay for downtime duration of 3 hours and 2886 min for 8 hours
- When the workload capacity of one rack is decreased during downtime, the average fill grades of the other racks become higher
  - Average rack fill grade of 16.35% for 3 other racks with always equal capacity and 17.49% for 3 other racks with a halved capacity of broken rack for 8 hours.
- When production is started sooner, the average rack fill grade increases
  - 7.78% when releasing production 4 hours before departure, 13.31% when releasing production 8 hours before departure
- When pallets cannot go directly to outputs, but are released for going to outputs with an offset from the departure, the average rack fill grade increases.
  - 18.57% with direct output release, 28.68% with output release 2 hours before departure
- When the production pool size decreases to 1, trips are produced in order of departure time
  - True

All hypotheses are confirmed when testing it with the model.

Hand calculations were also used for verification. The volumes returned by the script that converts the input data are aligned with the volume calculated from the original data. Additionally, the crane travel time between the actions of the crane in the event log is the same as calculated. Next to that, deterministic runs were performed with standard deviations of 0 for the stochastic processes enabling the replication of simulation steps by hand and verifying them.

The distributions used were also verified by extracting the resulting times from the event log. The trip selection, production time, loading start time offset and loading time distributions together with the data extracted from the simulation are shown in figure 7.2.



(a) Trip selection

(b) Production time

(c) Loading start offset

(d) Loading time

Figure 7.2: Realised samples from distributions used

Next to that, the following plots in figures 7.3, 7.4 and 7.5 were used for verification. In figure 7.3, it can be seen for example that the values for the number of possible crane tasks, the crane utilisation and the crane cycle time are in line with their expected values and no strange behaviour is observed. Next to that, the cranes are not utilised during downtime. Note that the outliers in the crane cycle times are because of occasional waiting time for another crane to finish its task to avoid collisions.

In figure 7.4 it can be seen that the gravity lanes and rack fill grades are always between 0 - 100 %, there are never more than 3 pallets at an input point and also here, the values are in line with their expected values and no strange behaviour is observed. Note that the gravity lanes fill grade drops during downtime since loading can still take place. Empty pallets can be taken off by hand instead of returning them by crane.

In figure 7.5, the combined fill grades and production and loading speed are as one would expect. Note that there are a lot of delays after the downtime of rack 1 ends since the trips that were stuck in an inaccessible part of the rack can be moved out of the rack by then. Also, there are fewer trips departed than produced at the end of the day since the simulation is stopped at the end of the target day, but by then there are already containers produced and in the rack of the next day.

For every component, unit tests were performed. When performing a unit test, a component or part of a component was separated and checked for a variety of scenarios and random seeds. All characteristics described in chapter 5 were checked and confirmed using the self-created event log both during the development of the model and afterwards. Additionally, the script that runs different simulation runs in parallel with varying random seeds and collects all results has been verified this way.

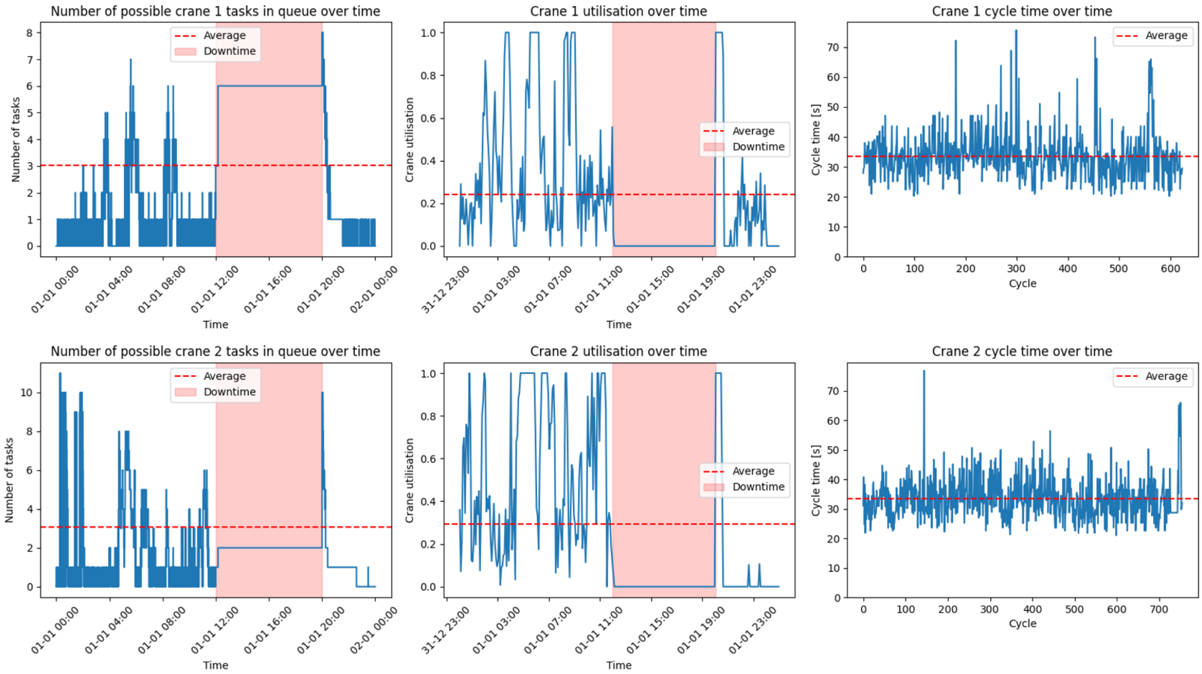


Figure 7.3: Cranes behavior

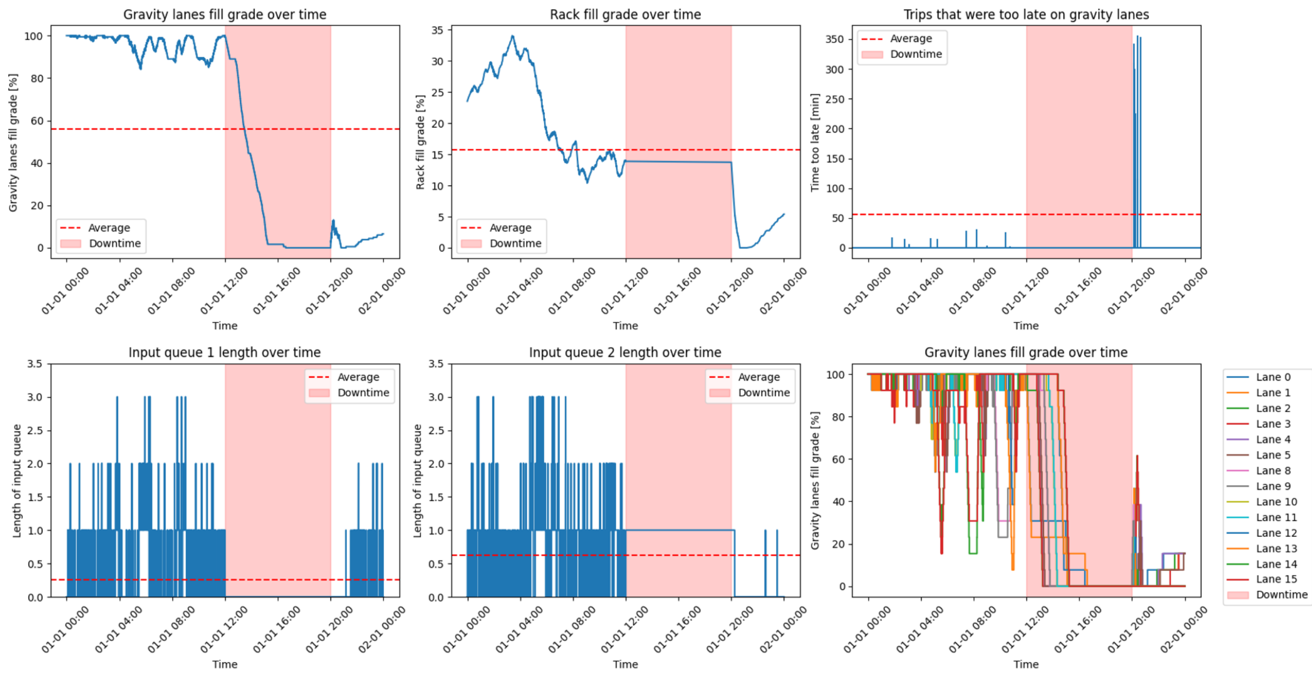


Figure 7.4: Rack behavior

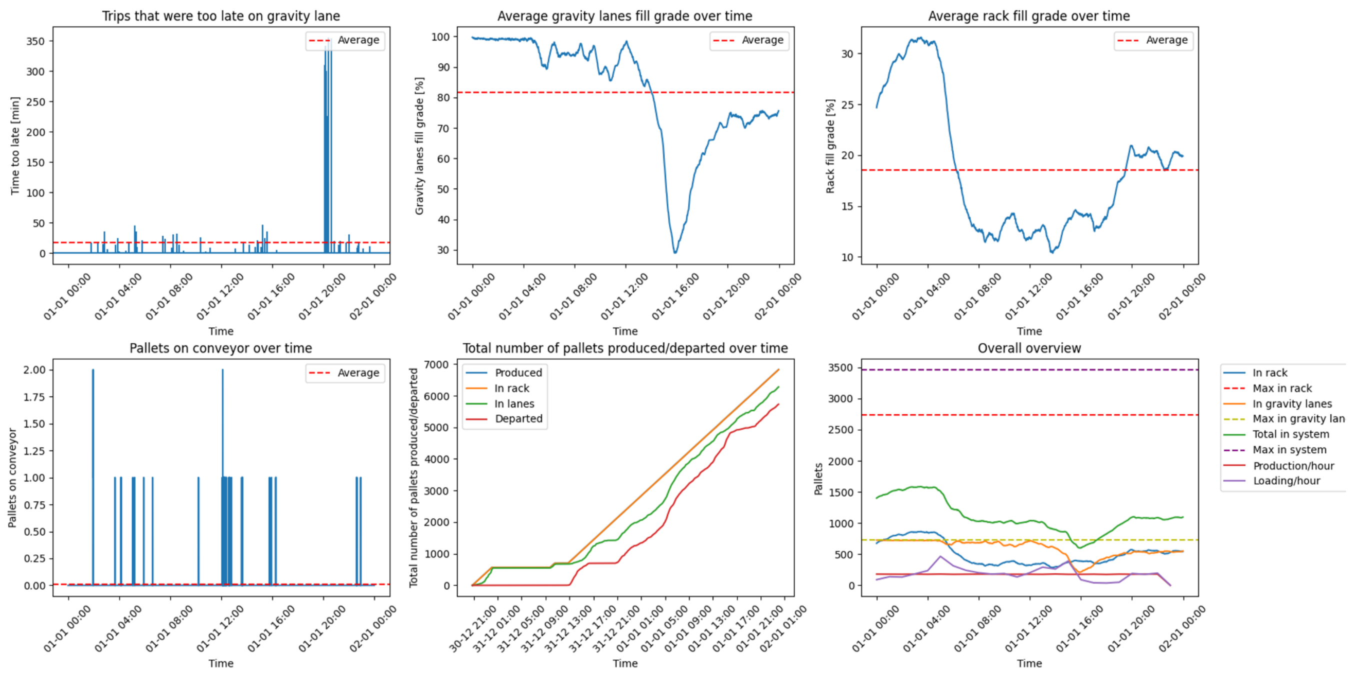


Figure 7.5: General behaviour

Lastly, sensitivity analyses were performed for the input parameters that were not exactly known but estimated. With this method, these parameters were altered to observe whether those changes caused variations in the results of the simulations. From this, it can be concluded whether those parameters need to be re-evaluated. The tested parameters were as follows:

- Rack switch-on delay
  - Value: 30 minutes after downtime stop
  - Alternate values tested: 0 or 60 minutes after downtime stop
  - Observation: No large variations in results
  - Conclusion: Accepted
- Loading start time before departure
  - Value: 30 minutes before departure with deviations from that based on a Gamma distribution with k=3 and theta=1.5, scaled with factor 250
  - Alternate values tested: Scale factor 100, means less extreme spread in downtime starts
  - Observation: Slightly elevated crane utilisation and fill grades due to trips being loaded closer to their original start time and therefore less trips falling outside of the target day. With a wider spread, more trips are loaded too early before or late after the target day
  - Conclusion: Variations are deemed acceptable because they are minimal
- Loading time per pallet
  - Value: Normal distribution with a mean of 69.23s and standard deviation of 5
  - Alternate values tested: Mean of 60s and 80s
  - Observation: No large variations in results
  - Conclusion: Accepted

- Production time per pallet
  - Value: Normal distribution with a mean of 18.09s and standard deviation of 2 (199 pallets/h)
  - Alternate values tested: Mean of 16s and 23s (225 and 156 pallets/h)
  - Observation 16s: Slightly elevated crane utilisation, increased average rack fill grade by 15%pt, max system fill grade elevated by 18%pt, slightly more pallets on conveyor
  - Observation 23s: Decrease in crane utilisation and fill grades, large increase in the number of delays due to production not being ready in time on peak day
  - Conclusion: Some larger variations were observed which do impact results. The calculation was double-checked, some adjustments were made and it was agreed with Jumbo that this is the best estimation
- Production pool size
  - Value: 30 trips
  - Alternate values tested: 20 and 40 trips
  - Observation: No large variations in results
  - Conclusion: Accepted
- Pallet pickup time crane
  - Value: 4.8s + 2.85s * depth
  - Alternate values tested: 4s + 2s * depth, 6s + 4s * depth
  - Observation 4s + 2s: Drop in average crane utilisation of 9 %pt, slightly less delays and pallets on conveyor
  - Observation 6s + 4s: Increase in average crane utilisation of 12 %pt, slightly more delays and pallets on conveyor
  - Conclusion: Cycle time is influenced by this which influences other KPIs, it does impact results to a certain extent, but double checked the calculation again and agreed with Jumbo this is still the best estimation
- Pallets allowed to move to output
  - Value: As soon as possible
  - Alternate values tested: 2 or 5 hours before departure
  - Observation 2 hours: Average gravity lane fill grade halved from 90 % to 45%, rack fill grade increased by 12 %pt.
  - Observation 5 hours: Slight decrease in average gravity lane fill grade and an increase in rack fill grade
  - Conclusion: Confirmed with Jumbo that the setting they will use is moving pallets to the outputs as soon as possible
- Production start
  - Value: As soon as possible
  - Alternate values tested: 5 or 10 hours before departure
  - Observation 5 hours: Decrease in crane utilisation of 8 %pt, average gravity lane fill grade halved from 90% to 45%, average rack fill grade dropped from 25% to 8%, major increase in delays since production is not ready in time
  - Observation 10 hours: No large variations in results increased since it is closer to original lead times in combination with workload
  - Conclusion: In reality, Jumbo would always like to start as early as possible so the value is not changed

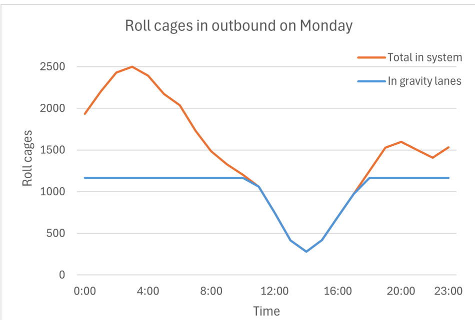
- Number of pallets that can be in a broken buffer for a trip to still be rerouted
  - Value: 10 pallets
  - Alternate values tested: 5 or 15 pallets
  - Observation 5 pallets: Number of pallets that can be rerouted drops significantly from ±130 to ±50 leading to more failure lane unloads, apart from that, no large variations in performance
  - Observation 15 pallets: Similar performance to original value, there are few trips with over 10 pallets in the broken buffer that are eligible to be rerouted for the scenarios tested
  - Conclusion: Although lowering this value does influence the amount of added manual work, overall buffer performance is not affected much and a value of 10 pallets is still desired

7.2. Validation

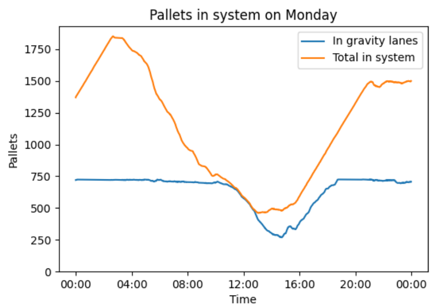
Since the system is not operational yet, no real-world data exists which can be used to validate the simulation model. Additionally, no comparable system could be found from which data was available. Therefore, the model was validated with experts on the system, an analytical model and design data.

The model with its assumptions and simplifications has been discussed with experts on the system and it was agreed upon that the simulation model sufficiently represents the system based on the available knowledge of how the system was designed by Witron and how the system will be operated by Jumbo.

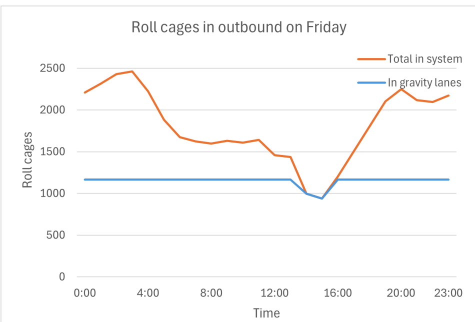
Within Jumbo, an analytical model was developed to track the predicted production throughout the day based on desired shipping volumes. This model also predicts the fill grades of the outbound system. These fill grades throughout the day can be compared to the fill grades resulting from the simulation model. Because the peak day in the peak week is an artificial dataset, only the first and second datasets which represent Monday and Friday can be compared. Some of the assumptions of the simulation model were altered to match the assumptions of the analytical model. The predicted fill grades of the outbound system of this analytical model can be seen in figure 7.6a and figure 7.6c. The fill grades resulting from the simulation model can be seen in figure 7.6b and figure 7.6d.



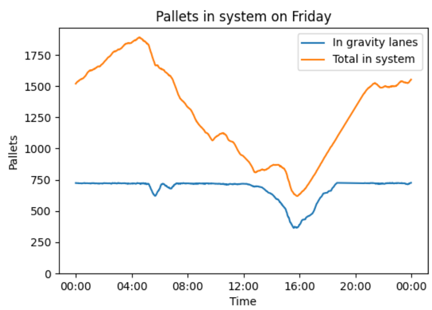
(a) Analytical model Monday



(b) Simulation model Monday



(c) Analytical model Friday



(d) Simulation model Friday

Figure 7.6: Fill grades of analytical model compared to simulation model

There is a slight mismatch since the analytical model does not take the crane operations and loading time into account. This can be seen in the maximum of the gravity lanes fill grades for example. In the analytical model, it is assumed the roll cages move to the gravity lanes directly, while in the simulation model, it takes time for a new pallet to be moved to the gravity lane after one was taken out, which is more representative of the real system. Also, in the analytical model, all roll cages of a trip leave the system at departure time while in the simulation model, the pallets leave the system gradually starting 30 minutes before departure time. The mismatch in the volumes is because of the different assumptions of the analytical and simulation models, such as rounding the number of roll cages and production speed is assumed constant in the simulation model, while it depends on the remaining workload in the analytical model. Also note that there are two roll cages on one pallet, and sometimes there is an empty roll cage on a pallet. However, it can be seen that the overall shapes of the graphs correspond, suggesting that the evolution of fill grades throughout the day is logical.

Most physical system parameters such as rack dimensions and crane speeds are exactly known, but the pickup time of a pallet was not exactly known and estimated. Based on data from another crane in the warehouse which takes 7.65s to pick up a pallet 1-deep and 10.5s to pick up a pallet 2-deep, the formula $t = 4.8 + 2.85 * depth$ was determined. To validate the overall crane cycle time which takes into account the travel time and pickup time, a design figure of Witron was used. This figure shown in figure 7.7 represents the average crane cycle which is a dual-command cycle. The crane is said to perform 50 dual-command cycles per hour. This means that this cycle where at P1 a pallet is picked up and dropped at P2 and another pallet is picked up at P3 and dropped at P4 should take 72 seconds. The distances have been extracted from the figure and the overall cycle time was calculated by hand with the formulae used in the model. This calculation resulted in a cycle time of 72.35s which is very close to the design value and thus considered valid.

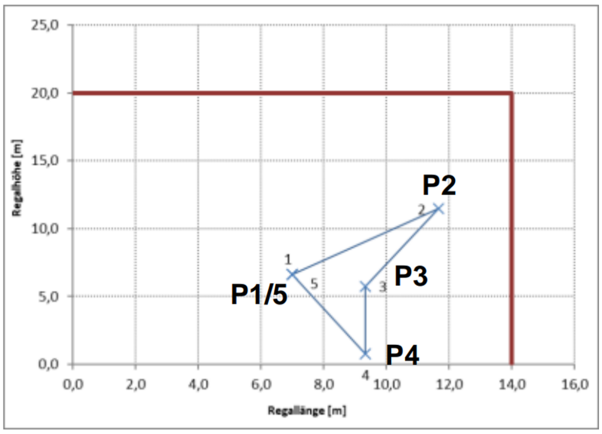


Figure 7.7: Average dual-command crane cycle

In conclusion, this chapter verified and validated the simulation model and thereby contributed towards the answer to sub-question 5.

8
Experiments

With a verified and validated model, the experiments can be evaluated with the developed simulation model. This chapter discusses the experiment setup, scenarios that will be evaluated and alternative configurations that will be tested.

8.1. Setup

As part of the experimental plan, the initialisation, run length and number of replications of the simulation need to be determined.

8.1.1. Initialisation

To create a realistic scenario, the AS/RS should not start empty, but should already be partially filled when production for the target day starts since that influences the rack locations where the new pallets get appointed. To consider this, the dataset includes all trips that leave after the first trip of the target day is released. Similarly, all trips that are released until the last trip of the target day departs are included to make sure that also pallets for the next day enter the rack which impacts fill grades and crane usage. In figure 8.1, the warm-up period for the simulation can be seen for a Friday.

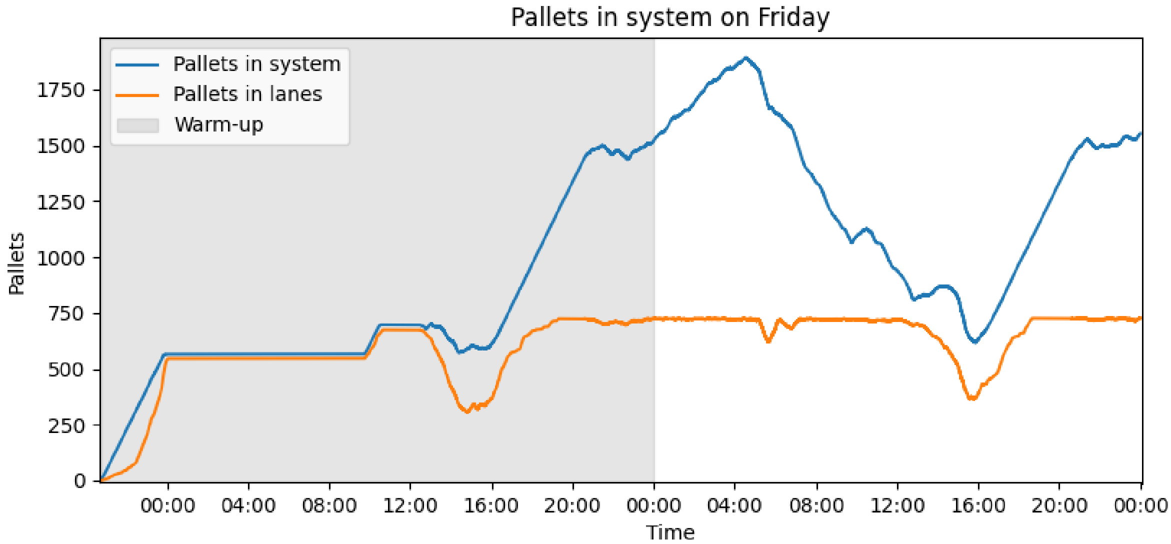


Figure 8.1: Warm-up period of simulation on Friday

8.1.2. Run length

It was decided to evaluate the KPIs during one full production day. Most days follow similar patterns regarding release times of orders and departure schedules, resulting in a similar development of fill grades and utilisation throughout the day. Because of this repetitive pattern, simulating more than one production day does not bring any additional insights.

From, among other things, the crane utilisation, it can be deduced that the conveyor, cranes and rack recover from downtime within a few hours as can be seen in figures 8.2 and 8.3. Note that this scenario is the busiest day with the experiment where both cranes are down and policy 0, taking no action, is applied resulting in the worst-case scenario. Therefore, it is not required to simulate for a longer time after the downtime stops. The latest downtime start is at 13:00 and the longest downtime duration is 8 hours, resulting in the last downtime stop being at 21:00. This leaves 3 more hours of simulated time which is sufficient to evaluate the effects of downtime after the downtime has stopped.

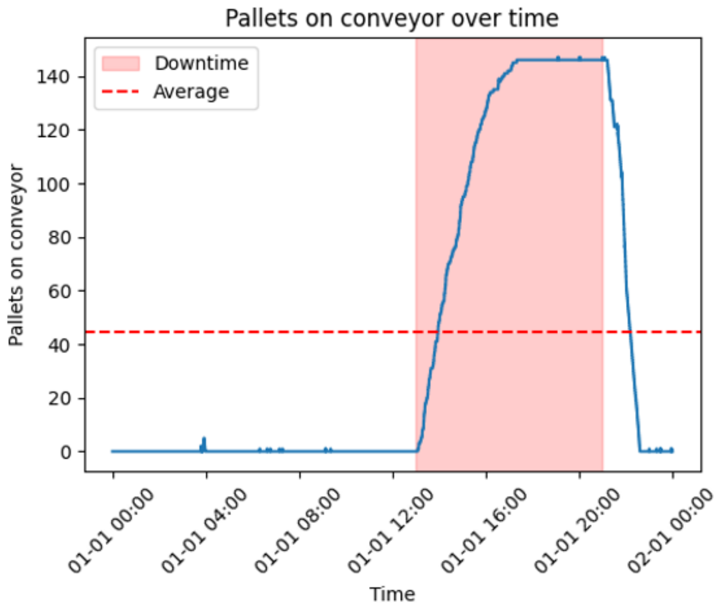


Figure 8.2: Number of pallets on conveyor during and after downtime in the worst-case scenario

8.1.3. Number of replications

Because some of the components contain stochastic elements, simulation runs with varying random seeds can produce varying results. To make sure the results are not influenced by the random seed, the simulation should be run a multitude of times to get accurate average results. The number of replications needed was determined by setting an allowed error and desired confidence interval for each KPI. The formula used to calculate the required number of replications is $n = (\frac{Z*\sigma}{E})^2$ where n is the number of replications, Z is the Z-score corresponding to the desired level of confidence, σ is the standard deviation and E is the allowed margin of error.

The desired confidence interval is set at 95% which corresponds with a Z-score of 1.96. The standard deviation results from the different simulation runs. The allowed margin of error is determined for each KPI extracted from the simulation and is shown in table 8.1. The allowed margin is based on the mean value of the KPIs returned from test runs and deciding which deviation from that mean would result in a different score. Because no exact numbers need to be extracted but the KPIs are purely used for comparison against each other, these margins of error are less strict reducing the overall computation time needed. The model keeps adding replications until the set criteria are met for each KPI, or if the number of replications exceeds 200 to limit computational time. The number of replications needed is mostly within 50-100 replications.

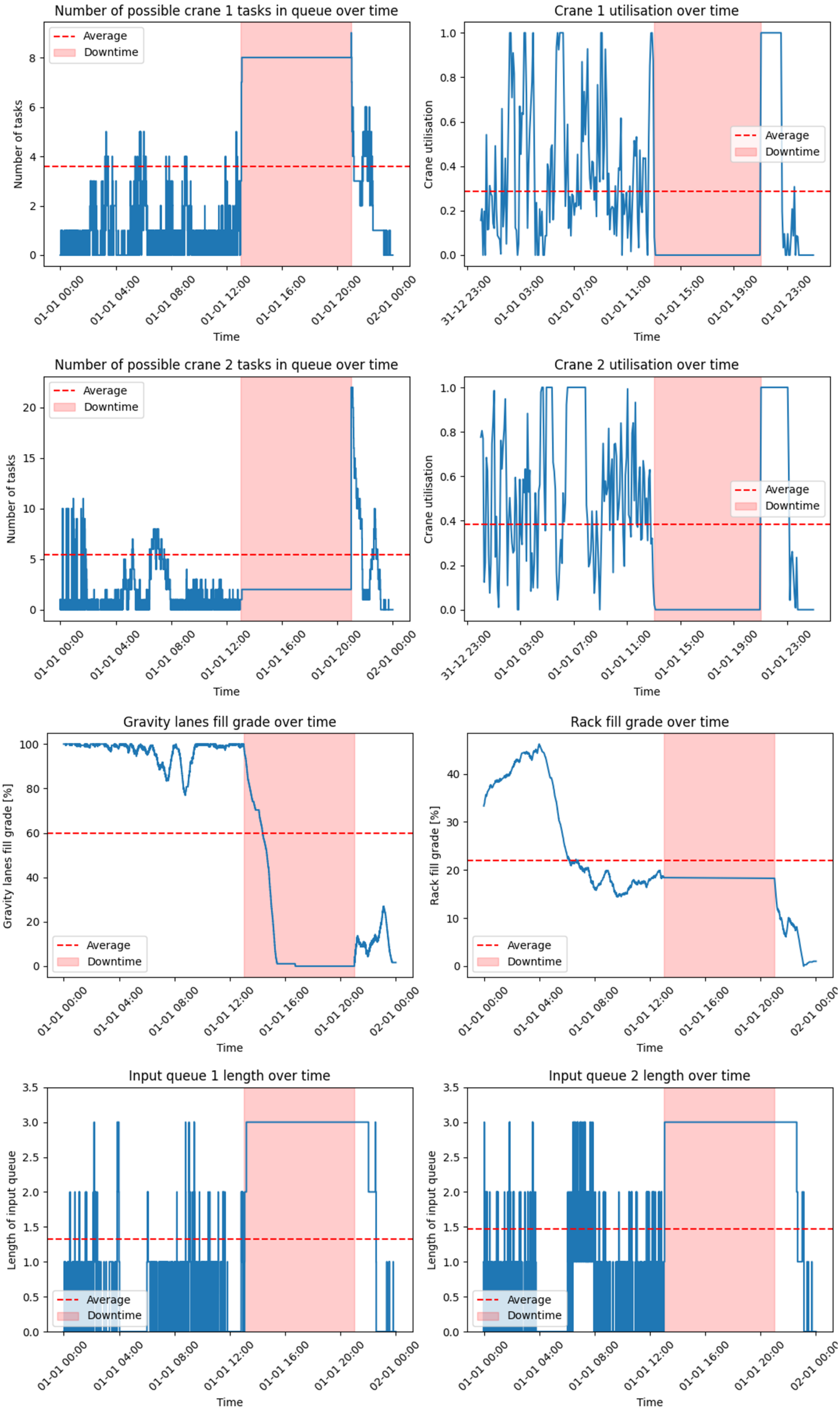


Figure 8.3: KPIs of cranes and rack after downtime on peak day

KPI	Allowed error
Aggregate trip loading delay [min]	20 min
Number of trips with delay [trips]	1 trip
Average number of pallets on conveyor [pallets]	0.1 pallets
Maximum number of pallets on conveyor [pallets]	1 pallet
Total number of failure lane unloads [pallets]	1 pallet
Maximum number of failure lane unloads for one rack [pallets]	1 pallet
Maximum crane utilisation [%/hour]	2%pt
Longest continuous utilisation [min]	5 min
Maximum system fill grade [%]	1%pt
Minimum margin between departure times on docks [min]	1 min
Maximum average fill grade since downtime stop [%]	1%pt
Average crane utilisation since downtime stop [%]	1%pt

Table 8.1: Allowed errors for each KPI

8.2. Scenarios

In general, two experiments are conducted, which are the experiment where one of the two cranes within a rack is down and the experiment where both cranes within a rack are down. For each experiment, the policies will be tested for different scenarios to be able to see if the best-performing policy changes for different scenarios.

There are three different input datasets which represent three different days, an average day in an average week, a peak day in an average week and a peak day in a peak week. During each day, it is determined that a downtime start time at 05:00, 07:00 and 13:00 will be tested for both experiments. These start times were chosen in order to cover the main production and loading peaks of the day. Also, both scenarios where production was already started at the downtime start time and where there was no production are included. This is necessary in order to take into account that sometimes a decision needs to be made about what to do with pallets that were already in production for a broken buffer.

Downtime durations of 3 and 8 hours will be tested. The downtime start moments represent some of the busier moments of the day, but with a longer downtime duration, also a quieter moment in the day is covered. A downtime duration longer than 8 hours will not be considered since the system comes to equilibrium after a longer duration of downtime. This is because no more trips would be appointed to the broken rack and the other racks have stabilised after receiving more workload. Note that in the experiment where one crane is down and the decision is made to keep operating it, this downtime duration is elongated with the times it takes to move the broken crane to and from the maintenance position.

For each day, it was determined when production should be stopped to ensure acceptable lead times between production being ready and the departure time while minimising the time that production is stopped during the day.

For the first dataset, it was determined that during the previous day, trips for the target day until a departure time of 09:30 will be produced resulting in pre-production until ±22:30 the day before, with production stops between ±10:00 and 12:45. This was necessary to guarantee acceptable lead times with a minimum lead time of ±1:30h.

For the second dataset, it was determined that during the previous day, trips for the target day until a departure time of 10:30 will be produced resulting in pre-production until ±00:45 with production stops between ±10:30 and 12:45. This was necessary to guarantee acceptable lead times with a minimum lead time of ±1:40h.

For the third dataset, it was determined that during the previous day, trips for the target day until a departure time of 12:00 will be produced resulting in pre-production until ±04:00 with no production stops. The limiting factor here was not the lead time but producing ahead so that there were no production stops which resulted in a minimum lead time of ±2:40h.

The number of different scenarios has been limited to reduce the needed computational time. This results in 18 scenarios per policy per experiment. Both experiments have 5 policies to test, which results in 144 scenarios in total. To deal with this, a script has been written that runs all scenarios at once per experiment per day so that the simulation only needs to be started a few times and can run in the background.

The rack in which the cranes break down was selected to simulate a worst-case scenario. Under normal circumstances, crane 1 of rack 3 seems to be the busiest one so that is the one that breaks down in experiment 1. Overall, rack 2 seems to be the busiest one so that is the one where both cranes break down in experiment 2. Even though the trips are divided evenly across all racks, the input data determines which rack and specifically which part of the rack is slightly busier because of the number of pallets per trip and the loading sequence within the trips that differ.

8.3. Alternative configurations

When executing the first simulations it was observed that the system at Jumbo has significant overcapacity under normal circumstances according to the model. Therefore, it is interesting to see if the best-performing policy changes under different circumstances. For example, when the crane availability or storage space is limited or when order lead times are shorter which could occur when Jumbo pushes the limits of the system more and starts to produce more. This can be reenacted by lowering the crane speed, decreasing the rack size and using the model option to release production with an offset from the departure time of the trip. The rack size cannot be limited too far since relocations are not taken into account and the model will stop when relocations need to happen. Jumbo already wants to avoid this situation by not filling the racks over 80%, but it can be investigated from which reduction in rack size relocations are needed, which lowers efficiency. These alternative configurations will not be tested as extensively as the original experiments, but some general simulation runs will provide additional insights.

In conclusion, this chapter described the experiment setup and thereby contributed to the answer to sub-question 5.

9
Results

After running the experiments specified in the previous chapter, the results can be compared. The results will be split for each day for each experiment. An overview of the quantitative results can be found in appendix D. For each day, the most interesting results will be highlighted and explained in this chapter, thereby answering sub-question 5.

9.1. Scoring

The resulting KPIs for each policy in each scenario will be compared with reference values. Within the minimum and maximum reference value, the policy is scored from 1 - 10 for each KPI.

When a higher numerical value for a KPI represents better performance, the formula used to calculate the score is: $score = 10 - \frac{x - x_{ref,min}}{x_{ref,max} - x_{ref,min}} * 9$

When a lower numerical value for a KPI represents a better performance, the formula used to calculate the score is: $score = \frac{x - x_{ref,min}}{x_{ref,max} - x_{ref,min}} * 9 + 1$

Where x is the numerical value of the KPI, $x_{ref,min}$ is the minimum reference value and $x_{ref,max}$ is the maximum reference value. The reference values are based on a combination of the KPI values for the scenario with no downtime and the difference between the KPI values resulting from the policies across both experiments. When the scores for each policy are very close for example, the reference value is determined in such a way that the scores are representative of the performance of the policies. The KPIs for all days when no downtime occurs can be seen in appendix D.1.

9.2. Experiment 1: One of two cranes within an AS/RS down

In this experiment, a decision has to be made about what to do with the workload sent to that AS/RS in case the decision is made to keep operating the remaining crane. The policies vary the workload from the original 100% to a workload of 0% compared to the original workload which would be beneficial in case the remaining crane cannot handle the full workload.

9.2.1. Average day in an average week

During this day, the performances of all policies are very similar as can be seen in appendix D.2.1. This is due to the lower volume handled on this day with the cranes being used just a quarter of the time and an average system fill grade under 30%. The maximum number of pallets on the conveyor is similar for all policies, and at around 5 pallets, this does not cause issues. Crane usage is ordinary, and the margin between departure times on docks is solid. The maximum system fill grade seems to elevate slightly with a lowered workload capacity of the rack with the broken crane but is still very moderate at ±50%. The other buffers can handle the extra workload under policy 4, 0% capacity, but it is slightly less robust and resilient in comparison to the other buffers.

The generated scores averaged over the four scenarios regarding downtime start moment and duration can be seen in table 9.1. A noteworthy occurrence is the minimum margin between departure times being reduced from 2.5 hours to 2 hours with policy 0, 100% capacity, in some scenarios. With all other policies, this margin was maintained. This happens because with just one crane remaining, 5/14 gravity lanes are out of reach and new trips can only be planned in the remaining lanes raising the trip frequency per lane. This makes policy 0, 100% capacity, slightly less robust, however, 2 hours between departure times still seems robust enough.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	9,9	9,7	10,0	9,8	9,6
Upstream system interference	8	9,7	9,9	10,0	10,0	9,8
Added manual work	6	3,5	3,9	4,0	4,3	5,3
Robustness	4	9,4	9,7	9,8	9,7	8,2
Resilience	2	10,0	9,6	9,8	9,3	8,1
Overall		**8,5**	**8,6**	**8,8**	**8,7**	**8,5**

Table 9.1: Averaged scores experiment 1 day 1

When looking at the individual results per AS/RS, it turns out that although the fill grades after the downtime started are raised, the other AS/RS do not suffer from the added workload in policy 4, 0% capacity. This is because their workload increases by just 30-60 pallets on this day. Additionally, shifting workload to the fully functioning buffers seems to help the recovery of the broken crane after downtime. More interestingly, retaining a higher workload leads to more failure lane unloads. This is because more pallets are in production for this buffer, and when they arrive at the buffer while the broken crane was just repaired and is being moved back into action, they need to be unloaded via a failure lane.

9.2.2. Peak day in an average week

During this day, performances of all policies are also very similar as can be seen in appendix D.2.2. Although the volume handled on this day is higher, the remaining crane seems to be able to manage the full workload since there are not noticeably more delays. There are slightly more pallets that have to wait on the conveyor for the policies retaining more of the original workload, but the difference is marginal. Compared to the average day in an average week, as expected, the crane utilisation and fill grades are higher, but this does not cause any issues.

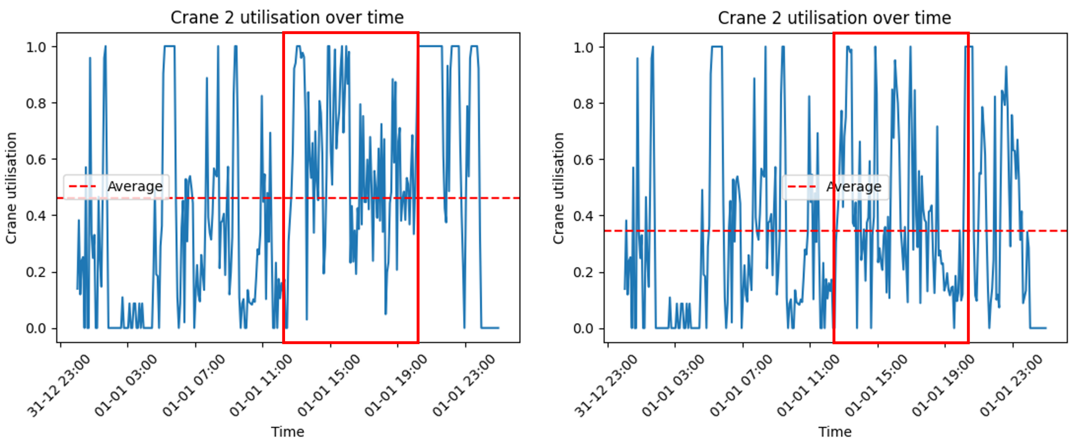
The generated scores averaged over the four scenarios regarding downtime start moment and duration can be seen in table 9.2. It should be noted that for some scenarios, the maximum number of replications was reached which could lead to small variations in the averaged KPIs.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	9,6	9,8	10,0	9,8	9,6
Upstream system interference	8	9,0	9,5	9,4	9,7	9,9
Added manual work	6	3,7	4,0	4,1	4,3	5,1
Robustness	4	8,0	9,1	9,6	9,9	9,2
Resilience	2	10,0	9,7	9,6	9,1	7,6
Overall		**8,1**	**8,4**	**8,6**	**8,6**	**8,6**

Table 9.2: Averaged scores experiment 1 day 2

The minimum margin between departure times on a dock is the same in the scenarios with a downtime start time at 07:00 and 13:00. This occurs because at the production start at 06:00, a large amount of trips is generated and added to the production pool. The trips which are generated during that time are the trips in the second departure peak between 13:00 and 16:00. The departure times of trips that are generated and divided over the buffers during the downtime, which starts at 07:00 or later, are more spread out, leading to a larger minimum margin between departure times per dock. With a downtime start at 05:00, there is a significant reduction in this margin between departure times per dock which is less robust. Retaining too much of the workload leads to a smaller margin, but shifting away too much of the workload also leads to a smaller margin in the other buffers.

When looking at the individual results per crane and rack, in figure 9.1 it can be seen that the crane usage of the remaining crane during downtime is considerably higher under policy 0, 100% capacity, compared to policy 3, 50% capacity, which is also reflected in the KPIs and scores. This results in the system being more robust under policies that shift more workload away from the partially broken buffer.



(a) Policy 0 - 100% capacity (b) Policy 3 - 50% capacity

Figure 9.1: Remaining crane utilisation experiment 1, day 2, downtime 13:00 - 22:15

9.2.3. Peak day in a peak week

During this day, the performance of the policies does differ significantly as can be seen in appendix D.2.3. After a downtime start at 05:00, the performances of the policies differ the most. Under policy 0, 100% capacity, the aggregate delay is considerably larger, there are more pallets waiting on the conveyor, and the longest continuous crane utilisation is over 4 hours as shown in figure 9.2. This occurs at the remaining crane after the downtime since all of the pallets coming for that buffer during the downtime have been placed in the area of the rack and gravity lanes that belong to that crane.

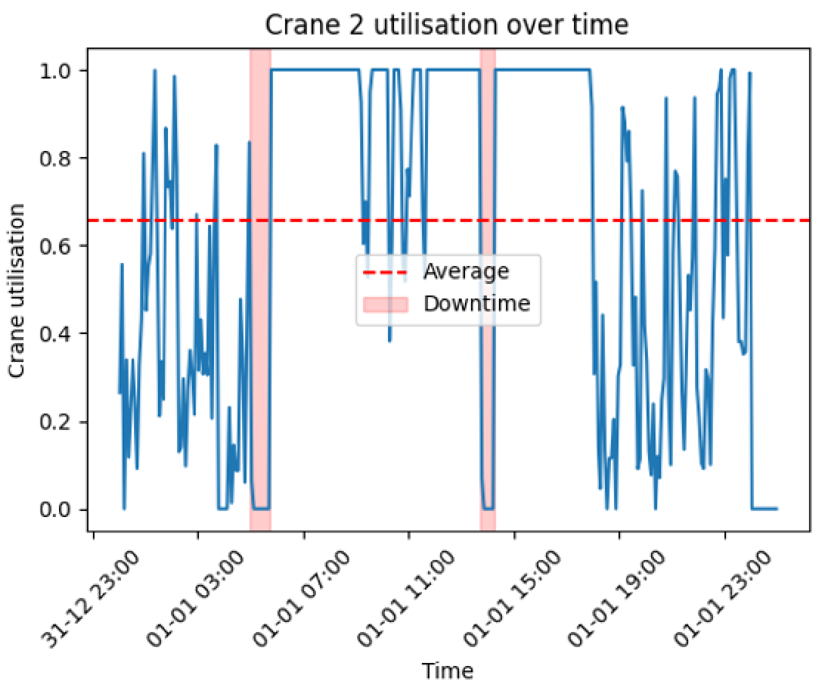


Figure 9.2: Remaining crane utilisation experiment 1 day 3

Next to that, the minimum margin between departure times is just 32 minutes in one of the scenarios which means that the next truck driver would start directly after the first one when the first driver would start loading on time and does not have any issues while loading. Under policy 3, 50% capacity, for example, the system performs way better in that scenario. In the other scenarios, the difference in performance is smaller which raises the scores for policy 0, 100% capacity, slightly but the difference is still noticeable.

The generated scores averaged over the four scenarios regarding downtime start moment and duration can be seen in table 9.3. Also here, for some scenarios, the maximum number of replications was reached which could lead to small variations in the averaged KPIs.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	9,3	9,7	9,8	9,9	10,0
Upstream system interference	8	8,9	9,3	9,5	9,5	10,0
Added manual work	6	3,2	3,5	3,6	4,0	4,9
Robustness	4	4,7	6,1	6,7	7,8	6,9
Resilience	2	9,2	9,7	9,6	9,0	7,2
Overall		**7,4**	**7,9**	**8,0**	**8,3**	**8,4**

Table 9.3: Averaged scores experiment 1 day 3

As reflected in the scores, policy 0, 100% capacity, performs worst since it causes more delays, causes more interference to the upstream systems, and is less robust and resilient. The margin between departure times on a dock is the smallest with downtime starting at 05:00 since around that time, most of the trips being produced still depart on this day. During later downtimes, most of the trips being produced are outside of the departure peaks leaving more room between departure times.

The maximum system fill grade is the same in all scenarios under all policies. This occurs because this maximum is reached before any of the downtimes start around 04:00 as can be seen in figure 9.3. During the other days, this is not the case since less volume has to be pre-produced and production stops earlier the night before.

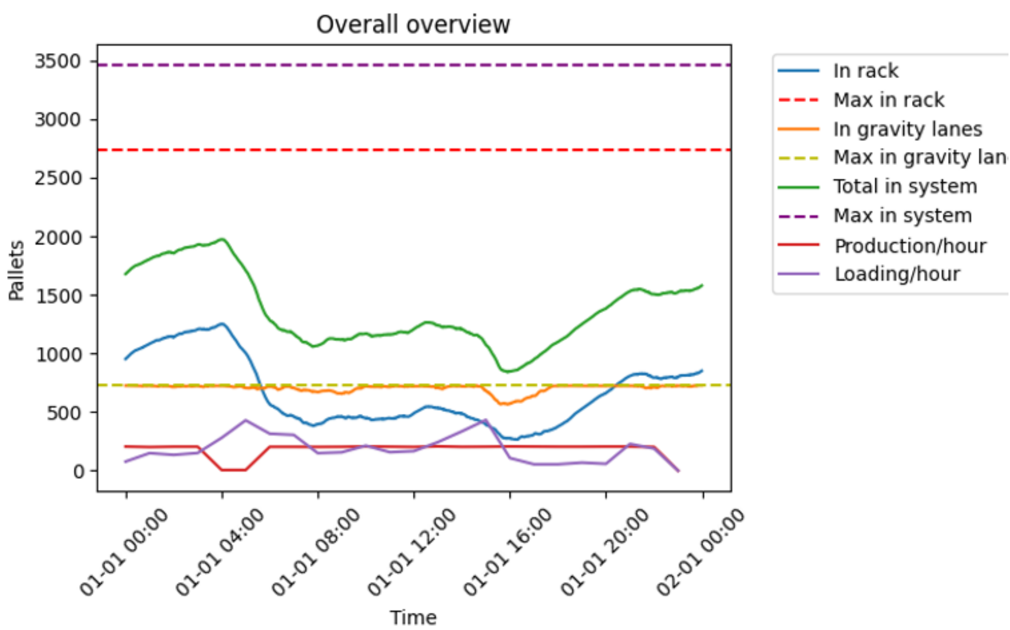


Figure 9.3: System fill grade experiment 1 day 3

All scores combined can be seen in table 9.4. It can be seen that the policy which retains the original capacity performs worst, although in most scenarios it still performs acceptably. As expected, the higher the overall workload during the day, the worse policy 0, 100% capacity, performs, but interestingly enough, during an average week, following this policy does not cause many problems in the specified scenarios. Policy 3, 50% capacity, seems to perform best since it is in the middle between shifting too much workload away to other buffers which makes them less robust and resilient, and keeping too much workload in the buffer with one functioning crane. The combined scores for policies 1-4 are very close and it seems like the higher the workload on a day, the more important it becomes to shift more workload away to other buffers. In systems with less overcapacity, this might not be the case and shifting much workload to another buffer might not be an option.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	9,6	9,7	9,9	9,8	9,7
Upstream system interference	8	9,2	9,6	9,6	9,8	9,9
Added manual work	6	3,5	3,8	3,9	4,2	5,1
Robustness	4	7,4	8,3	8,7	9,2	8,1
Resilience	2	9,7	9,7	9,6	9,1	7,7
Overall		**8,0**	**8,3**	**8,5**	**8,6**	**8,5**

Table 9.4: Averaged scores experiment 1 across all days and scenarios

9.3. Experiment 2: Both cranes within an AS/RS down

In the experiment where both cranes within an AS/RS are down, a decision has to be made about what to do with the pallets already in production for this now-broken buffer. They could be rerouted to another buffer and given a destination there, or be directly unloaded there which creates an additional decision to make about which buffer(s) to send the pallets to. Additionally, it is interesting to see the effects of the extra workload on the other buffers during the downtime. Since during the average day in an average week, there is no production at 05:00, this scenario is also included. If no pallets were already in production for the buffer that breaks down, no decision has to be made about what to do with those pallets.

9.3.1. Average day in an average week

The performance of the policies during this day can be seen in appendix D.3.1. For policy 0, where no action is taken, it can be seen that in all scenarios where production was started, over 100 pallets are waiting on the conveyor at the same time which causes significant problems and causes the upstream systems to have to shut down since there is nowhere to send the produced pallets to. The maximum number of pallets on the conveyor does not depend on the downtime duration since after the downtime starts, the capacity of that buffer is set to 0 and all new workload is divided over the other 3 buffers.

Between policies 1 and 3, which both unload all pallets destined for the broken buffer through failure lanes, no major differences are observed apart from the maximum number of failure lane unloads for a single rack and the maximum crane utilisation. Unloading more pallets at a single rack instead of spreading them out means that the pallets will arrive at the failure lane at a high rate, which means the containers need to be taken off of the pallets on the failure lane at a high rate. These containers need to be moved to the locations specified in section 6.4.4. When all containers exit the system at one location, eventually, one has to move these containers further away since the storage locations close by are full at some point, causing more manual work. When these containers are unloaded spread out, this is not the case. The maximum crane utilisation is higher under policy 1, unload at partner, compared to policy 3, unload spread out, since all of the pallets are unloaded through one buffer meaning that they all need to be moved from the input points to the failure lane while still performing the normal duties.

Between policies 2 and 4, which both reroute pallets to another buffer and give them a destination there, the main differences can be seen in the maximum crane utilisation, the minimum margin between departure times on a dock and the fill grades and crane utilisation after downtime stopped. Since the production that was already in progress for the broken buffer is rerouted to a single buffer, all extra workload is concentrated there causing elevated maximum crane utilisations and fill grades. Additionally, the minimum margin between departure times on a dock is lower there because of this.

When comparing the policies unloading the already scheduled workload via the failure lane and the ones rerouting them, it can be seen that failure lane unloads occur under all policies. This is because some of the pallets that were in the broken buffer will be unloaded via the failure lane once the buffer is back up if the rest of that trip was rerouted to another buffer for policies 2 and 4. Similarly, pallets of trips from which too many pallets were already in the broken buffer are also unloaded via the failure lane in another buffer. Next to that, the main difference is policies 1 and 3 causing more manual work than policies 2 and 4, but once this workload is unloaded, the recovery is better. The scores can be seen in table 9.5.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	5,3	8,4	8,1	8,4	8,3
Upstream system interference	8	1,0	9,2	9,5	9,8	9,6
Added manual work	6	10,0	1,0	6,0	3,7	6,0
Robustness	4	7,6	6,9	6,6	8,2	8,0
Resilience	2	8,1	8,2	7,5	8,2	7,9
Overall		**5,6**	**6,9**	**7,8**	**7,8**	**8,1**

Table 9.5: Averaged scores experiment 2 day 1

9.3.2. Peak day in an average week

The performance of the policies during this day can be seen in appendix D.3.2. Compared to the average day, under policies 2 and 4, there are fewer pallets unloaded via the failure lane even though the overall throughput of the day is higher. This occurs since the trip frequency per dock is higher, which means that less often a pallet of a trip in the broken buffer will already be in the gravity lanes because the previous trip would still be there. This allows that trip to still be replanned to another buffer.

Next to similar observations as described in the previous section, it can be seen that the minimum margin between departure times on a dock under policy 2, store in partner, is just 33 minutes when the downtime starts at 07:00. Additionally, under this policy, the maximum crane utilisations, system fill grades and the aggregate delay are elevated. With downtime from 13:00 - 21:00, the maximum system fill grade for 1 buffer is 74% which is high and approaching levels where relocations are required.

With a downtime of 3 hours, because of the production pool, not all pallets that were generated have been produced yet. This is why with a downtime of 8 hours, there are more pallets on the conveyor under policy 0, no action. In figure 9.4, it can be seen that the number of pallets on the conveyor still increases after 3 hours causing a downtime of 8 hours to result in more added manual work.

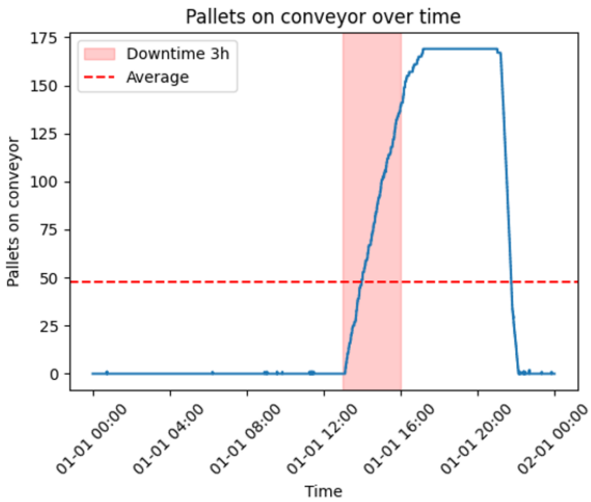


Figure 9.4: Pallets on conveyor over time Experiment 2, Day 3, Downtime 13:00 - 21:00, Policy 0, no action

The upstream system interference is similar under all policies. This means that the other buffers can process the pallets originally meant for the broken buffer fast enough while also keeping up with the newly generated added workload which is divided over 3 instead of 4 buffers. Policies 1 and 3 are more robust concerning the rack, but not concerning the crane since the pallets headed to the failure lane still need to be moved by the crane which influences the maximum crane utilisation. However, the average crane utilisation is lower since these pallets just need to be moved from the input to the failure lane once instead of first having to be moved from the input to the rack and then from the rack to the output. The scores can be seen in table 9.6.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	4,4	6,0	5,7	6,0	5,9
Upstream system interference	8	1,0	9,5	9,9	10,0	10,0
Added manual work	6	10,0	1,0	7,9	3,7	7,9
Robustness	4	7,7	7,3	5,0	8,7	8,3
Resilience	2	7,5	7,3	4,8	7,4	6,9
Overall		**5,3**	**6,2**	**7,1**	**7,0**	**7,8**

Table 9.6: Averaged scores experiment 2 day 2

9.3.3. Peak day in a peak week

The performance of the policies during this day can be seen in appendix D.3.3. When looking at the delays, policy 2, store in partner, seems to have slightly more delays in most scenarios, but not much. The delays do not differ much between policies since regardless of what happens with the pallets arriving after the downtime starts, the pallets stuck in the broken buffer dictate the delays. If the workload for the cranes in a buffer becomes too high, this should be observable when looking at the number of pallets waiting on the conveyor over time, but as can be seen in figure 9.5, even for the policy which concentrates the most workload on one buffer, policy 2, store in partner, there is no peak in number of pallets waiting on the conveyor since the downtime started.

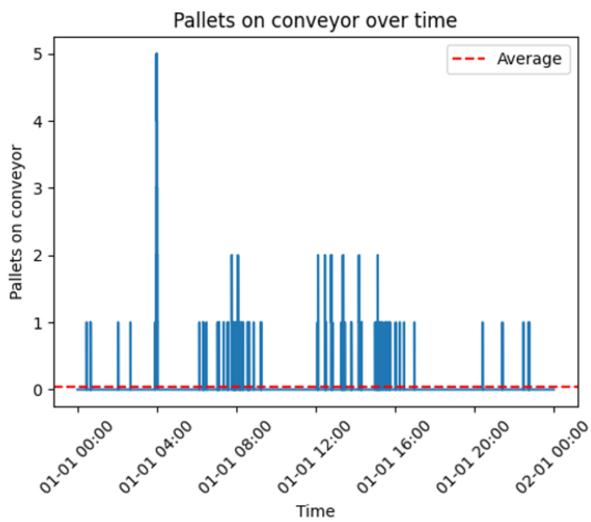
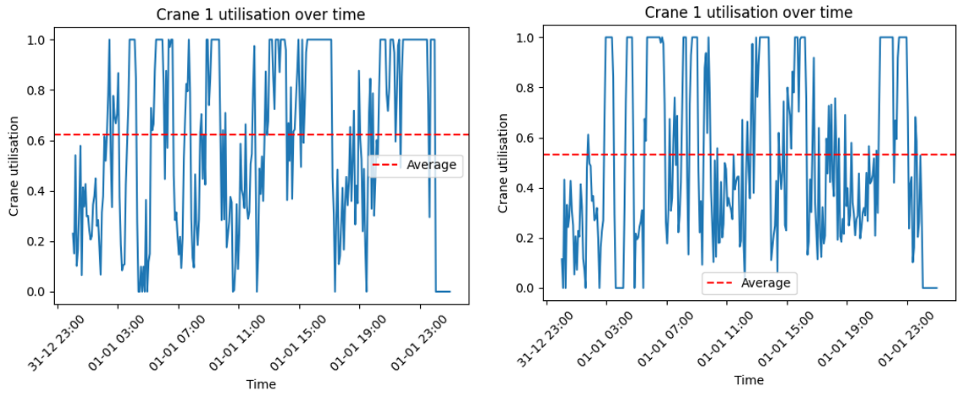


Figure 9.5: Number of pallets on conveyor, experiment 1, day 3, downtime 07:00 - 15:00, policy 2, store in partner

During this day, there are fewer direct unloads under policies 1 and 3 compared to the peak day in an average week. This happens as a result of the starting time of the downtime. During the average week, there is a slight dip in the production just before 13:00 which means that during that time, the production pool of 30 trips is not fully filled. Then, at 12:45, a new wave of orders is released. From these newly added orders, most pallets have not been produced yet resulting in a production pool with more pallets pending to be produced. On this day, the production pool was still filled at 12:45, but then with trips from which on average half of the pallets have already been produced, resulting in a production pool with fewer pallets pending to be produced.

Next to that, the maximum average crane utilisation for one crane since the downtime stopped exceeds 90% under policy 2, store in partner, with downtime from 13:00 - 21:00 as can be seen in figure 9.6, compared to under 70% for the similar crane in its mirrored buffer. This means the crane in the partner buffer is working almost continuously for the last 3 hours of the day which is not robust. It can be seen that spreading the workload over the other 3 buffers instead of just 1 is more robust. The only downside to spreading out the workload could be when the containers need to be replanned in or close to the original buffer later, but since Jumbo has not fully defined their replan process yet, this is not taken into account in this research.



(a) Busiest crane of partner buffer (b) Busiest crane of reference buffer

Figure 9.6: Crane utilisation experiment 2, day 3, downtime 13:00 - 21:00, policy 2, store in partner

The scores for the policies during this day can be seen in table 9.7 in which it can be seen that spreading the workload over the other 3 buffers instead of just the partner buffer performs better. Giving the pallets a destination in the other buffer is preferable since it saves extra manual work and the other buffers have sufficient overcapacity to handle this extra workload.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	5,0	5,8	5,7	5,9	5,8
Upstream system interference	8	1,0	9,4	9,9	10,0	10,0
Added manual work	6	10,0	1,0	8,2	3,6	8,2
Robustness	4	6,4	5,2	4,1	7,3	6,6
Resilience	2	6,8	6,7	3,8	6,8	5,9
Overall		**5,2**	**5,8**	**7,0**	**6,8**	**7,5**

Table 9.7: Averaged scores experiment 2 day 3

All scores combined can be seen in table 9.8. The performance of the policies does not differ much across days with respect to each other. The policies spreading out the workload always perform better and the policies storing the pallets in the AS/RS always perform better than directly unloading them since manual work is saved and there is sufficient overcapacity for the other buffers to handle the extra workload.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	4,9	6,7	6,5	6,8	6,7
Upstream system interference	8	1,0	9,4	9,8	9,9	9,9
Added manual work	6	10,0	1,0	7,4	3,7	7,4
Robustness	4	7,2	6,5	5,2	8,1	7,7
Resilience	2	7,5	7,4	5,4	7,5	6,9
Overall		**5,4**	**6,3**	**7,3**	**7,2**	**7,8**

Table 9.8: Averaged scores experiment 2 across all days and scenarios

9.4. Experiment comparison

When one of the two cranes breaks down, a decision needs to be made about whether to keep operating the remaining crane, or to stop both cranes. When it is desired to operate with a single crane, the broken crane needs to be moved to the side behind a maintenance fence in order to allow repairs while the other crane is in operation, and in order to make sure the other crane can operate in a larger range. Moving the broken crane to and from its maintenance position costs 75 minutes in total. During this time, the other crane cannot be operated, meaning that all pallets arriving at the inputs during that time cannot be handled, causing the need to unload them through the failure lane of the partner buffer to avoid congestion of the central conveyor.

To aid in this decision, both experiments can be compared since the same scenarios and reference values were used for performance evaluation. The best-performing policy while one crane was down is policy 3, which halves the rack capacity, while the best-performing policy while both cranes were out of operation was policy 4, which spreads the workload over the other 3 buffers. The performance metrics and scores comparing both scenarios and policies can be seen in tables 9.9 and 9.10 respectively.

		Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
	Remaining crane	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
Day 1	Operate	1312,3	41,4	0,03	5,16	66,9	39,3	90,2	54,2	50,0	151,0	36,8	43,0
	Stop	1374,2	42,6	0,03	5,54	42,6	32,7	93,6	63,7	53,4	150,7	40,6	46,0
Day 2	Operate	1462,3	57,5	0,02	3,93	83,2	46,8	92,6	61,6	54,8	120,5	39,8	47,7
	Stop	2055,9	58,7	0,01	3,61	31,5	17,1	93,8	62,8	58,0	99,5	44,6	51,5
Day 3	Operate	1902,9	57,6	0,04	4,54	77,9	42,8	99,4	108,3	57,7	88,5	40,0	52,7
	Stop	2682,4	62,0	0,02	4,03	20,8	15,0	98,0	110,5	58,6	73,2	45,3	60,0

Table 9.9: Performance metric comparison of best-performing policies for both experiments

	Remaining crane	Output delay	Upstream system interference	Added manual work	Robustness	Resilience	Overall
Day 1	Operate	9,8	10,0	4,3	9,7	9,3	**8,7**
	Stop	8,3	9,6	6,0	8,0	7,9	**8,1**
Day 2	Operate	9,8	9,7	4,3	9,9	9,1	**8,6**
	Stop	5,9	10,0	7,9	8,3	6,9	**7,8**
Day 3	Operate	10,0	9,5	4,0	7,8	9,0	**8,3**
	Stop	5,8	10,0	8,2	6,6	5,9	**7,5**

Table 9.10: Score comparison of best-performing policies for both experiments

From the above tables, it can be seen that across all studied scenarios, it is most beneficial to move a broken crane to its maintenance position and to keep operating the remaining crane, even though this slows down repairs. The most important reason for this is the difference in delays. When operation with one crane is continued, over half of the rack can still be serviced. With both cranes out of operation, all pallets in that buffer are stuck for the whole duration of the downtime and arrive at the outputs too late. Notably, stopping both cranes does reduce the amount of added manual work. This is a result of all pallets being unloaded through the failure lane of the partner buffer at moments when a crane is being moved to or from its maintenance position.

When looking at individual scenarios, there are scenarios where stopping both cranes performs better. For example with downtime from 05:00 - 08:00. In this scenario, there is no production at the downtime start. This results in no added manual work for the scenarios where both cranes are down since all newly generated pallets avoid this buffer. Also, fewer delays are observed during the average day. During the busier days, stopping both cranes does cause a larger aggregate delay for downtime from 05:00 - 08:00. Additionally, with shorter downtimes, the 75 minutes of added downtime when moving a broken crane might become too large with respect to the original downtime. To test this, experiments were simulated with 1 hour of repair time. In experiment 1, this results in a downtime of 2 hours and 15 minutes, while in experiment 2, the downtime remains 1 hour. The performance metrics for the experiments in these scenarios can be seen in table 9.11. From these metrics, it can be seen that there is a limit in downtime from when moving the broken crane becomes unattractive. Logically, more than doubling the downtime for the broken crane and stopping the remaining crane for 75 minutes does not weigh up to stopping both cranes for the original downtime duration of 1 hour.

		Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
	Remaining crane	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
Day 1	Operate	753,8	40,6	0,03	5,09	58,5	47,4	88,4	55,0	48,2	140,3	29,5	39,0
	Stop	695,6	40,6	0,03	5,69	25,6	15,8	91,2	56,9	48,8	150,7	31,2	39,9
Day 2	Operate	1041,2	57,2	0,02	3,98	66,0	50,1	93,0	61,6	54,6	121,0	33,2	43,3
	Stop	976,2	56,2	0,01	3,55	22,7	14,3	92,6	58,6	54,8	112,0	36,4	46,4
Day 3	Operate	1127,2	56,9	0,03	4,58	68,8	50,5	96,7	89,1	57,7	85,3	35,7	51,7
	Stop	1012,6	56,8	0,02	3,84	19,3	14,7	95,3	77,1	57,7	85,7	38,8	54,8

Table 9.11: Performance metric comparison of best-performing policies for both experiments with 1h of repair time

Generally, factors that make sure that stopping both cranes becomes favourable include:

- Low rack fill grades: less pallet stuck in rack when both cranes stopped
- Short downtimes: adding 75 minutes of downtime to move the crane is not worth it
- No generated production at downtime start: newly generated pallets will avoid broken buffer and less failure lane unloads necessary

9.5. Sensitivity analysis KPI weights

To determine the sensitivity of the ranking of the policies to the KPI weights, a sensitivity analysis was performed. An allowed deviation from the chosen weights by 1, 2 or 4 were tested and the ranking of the policies was determined for all of the possible weight combinations within the specified margins. The ranking of the policies for an allowed weight deviation of one can be seen in table 9.12. The rankings for margins of 2 and 4 can be seen in appendix D.5. As it turns out, the best-performing policy per experiment stays the same for >95% of weight combinations with a margin of 1. When comparing both experiments, operating with a single crane performs better than stopping both cranes with all weight combinations. Even for an allowed margin of 4, which means the weights can be either 4 less or 4 more than their original value, the best-performing policy stays the same with >85% of weight combinations. Similar conclusions can be drawn from the averaged scores across all weight combinations.

	Split ranking [%]					Combined ranking [%]										Avg
Policy	1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	score
One crane down - 100% capacity	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	68,5	31,5	0,0	0,0	0,0	0,0	7,93
One crane down - 75% capacity	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	0,0	0,0	8,28
One crane down - 67% capacity	0,0	24,7	75,3	0,0	0,0	0,0	24,7	75,3	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,42
One crane down - 50% capacity	95,7	4,3	0,0	0,0	0,0	95,7	4,3	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,55
One crane down - 0% capacity	4,9	71,0	24,1	0,0	0,0	4,9	71,0	24,1	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,47
Both cranes down - No action	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	100,0	5,36
Both cranes down - Unload at partner	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	100,0	0,0	6,29
Both cranes down - Store in partner	0,0	72,8	27,2	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	72,8	27,2	0,0	0,0	7,33
Both cranes down - Unload spread out	0,0	28,4	71,6	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	28,4	71,6	0,0	0,0	7,23
Both cranes down - Store spread out	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	31,5	68,5	0,0	0,0	0,0	0,0	7,86

Table 9.12: Ranking of policies with all possible weight combinations with an allowed weight deviation of 1

9.6. Additional observations

The results from the experiments have been analysed and some alternative settings were used to gain additional insights into the system:

- Because of the way the loading start time was modelled, there will always be delays regardless of the system performance since in reality, it will also occur that drivers are too late which impacts the system, for example when the margin between departure times on a dock becomes too small.
- More delays occur with downtime from 05:00 - 13:00 since most incoming pallets and pallets already in the rack belong to a departure peak. With later downtimes, there is a larger fraction of incoming pallets and pallets already in the rack from which the departure times are already more spread out.
- The maximum delay for one trip is dictated by the pallet that is stuck in an unreachable area of the rack with the earliest departure time and does not change with varying policies.
- One of the two cranes is always busier than the other one because of the asymmetry of the racks.
- On average, when both cranes within a buffer break down, there are between 110-170 pallets in production for this buffer about which has to be decided where to move them depending on the day and downtime start time.
- Because of the overcapacity, there seem to be little to no downsides in shifting workload from a (partially) broken buffer to the other buffers. There should be no need to unload pallets originally meant for a broken buffer to a failure lane when they can be replanned for another buffer which saves manual work.
- When downtime for a whole buffer starts at a moment when there is no production, the performance for all policies is the same since this buffer will be avoided when production starts.
- On a quieter day, the trip frequency per dock is lower, which means that more often pallets of a trip in production will already be in the gravity lane making it impossible to reroute the trip to another buffer during full downtime.
- During longer downtime, while not changing the capacity of the rack with 1 broken crane, that crane utilisation remains elevated even after the downtime stopped, since all of the incoming pallets during the downtime are stored in the area that is serviced by that crane.
- Even during the peak day in the peak week, there is enough capacity on the floor to store the containers that will be unloaded via the failure lanes as a result of the downtime of cranes.
- Having the failure lane in the middle has the advantage that both cranes can reach it, but the disadvantage is that this can cause interference with the other crane when a lot of pallets are unloaded via the failure lane.
- The tested delivery schedule does not cause problems for the buffers under normal operation.

- In the worst-case scenario, with both cranes down under the policy that unloads all pallets at the partner buffer, a maximum of ±50-60 pallets arrive at the failure lane per hour, meaning 2 containers have to be taken off the failure lane every minute at peak moments. This is regardless of the overall workload on the day but depends on the production pool size at the moment downtime starts. This value could be higher in case the production pool behaves differently and produces pallets faster.
- Reducing the maximum crane speed to 1 m/s, which represents a scenario with less crane overcapacity, leads to increased delays in extreme situations. For instance, when one crane is down and the buffer maintains its original capacity on the busiest day, or when all pallets are rerouted to the partner buffer with both cranes down on the busiest day. However, during more typical scenarios, the impact is minimal due to the initial overcapacity and lead times.
- Without downtime and issues, according to the model with its assumptions, when reducing the rack height and thus overall size, relocations start to occur with a halved rack height of 4 for all days while the original rack height is 8 rack locations. It is expected that relocations will occur earlier with a higher production speed, longer lead time or if pallets cannot go to the gravity lane as soon as possible since this increases the rack fill grade. A reduced rack size, and thus less overcapacity in that regard, would improve the performance of the policies diverging workload in experiment 1, and the ones unloading the pallets via the failure lane in experiment 2. The performance deteriorates for the policy keeping the capacity original in experiment 1, and the policy rerouting all pallets to the partner buffer in experiment 2.
- Under normal circumstances, more delays start to occur when pallets are allowed to go to the gravity lanes less than 1 hour before departure time instead of as soon as possible on an average day. On the peak day in an average week and on the peak day in a peak week, this is 1.5 hours.
- Decreasing the lead time from production release to departure time does not cause issues for the buffers. As long as the pallets arrive at the buffer in time, the buffer can sort and export them relatively quickly. Moreover, reducing the lead time will lower the average rack fill grade considerably making the operations within the buffer more efficient.
- It takes the buffer in which one crane was down approximately 1 hour for the cranes to handle all their pending tasks and recover in an average week and up to 2 hours on a peak day in a peak week in case the full capacity is kept. In case capacity is set to 0, this time reduces to ±15-30 minutes. In case both cranes were down and no pallets are waiting on the conveyor, it will take those cranes up to 30 minutes to handle their pending tasks and recover in an average week and up to 1 hour on a peak day in a peak week.
- The minimum margin between departure times on a dock is influenced most when downtime occurs at moment when trips from departure peaks are produced.
- When pallets are not sent to the failure lane of the partner buffer while a crane is being moved behind the maintenance fence, but wait on the conveyor, there is a peak of ±40 pallets waiting on the conveyor for the worst-case scenario during all days.

In conclusion, sub-question 5 can be answered for both experiments. In case one of the two cranes is down, a reduction in workload can reduce performance losses, but a too-large reduction in workload for this AS/RS increases the workload for the other AS/RS thereby making the system less robust. In case both cranes within an AS/RS are down, taking no action will cause the upstream system to have to come to a standstill. Storing the pallets that were already in production for the now broken AS/RS in the neighbouring AS/RS works on quieter days, but causes problems on peak days. Spreading them out over the other AS/RS performs better. Directly unloading them could prove beneficial in case the other AS/RS cannot handle the extra workload, but it turns out this is not necessary because of the overcapacity of the system. Nevertheless, unloading them spread out over the other buffer proves more efficient than unloading all pallets at the neighbouring buffer.

10
Discussion

In this chapter, the limitations of this study and the generalisation of the results will be discussed. Since there is no literature on downtime within AS/RS, work distribution across parallel AS/RS or Order Consolidation Buffers, no comparison with existing literature can be made.

10.1. Limitations

The model was developed to represent the system at Jumbo in detail, however, as in any simulation model, assumptions and simplifications had to be made which could influence the results generated by the model. Through the sensitivity analysis, it was determined that variations in most approximated values have little influence on the results. However, for example, the production speed does have a significant influence. If in reality, the production speed deviates significantly from the approximated value, results could be different. With higher production speeds, the average fill grade of the system rises and the crane utilisation is less spread out over the day. This will make policies which perform well because of the overcapacity of the system perform worse, such as policies retaining more of the original workload in experiment 1, and policies concentrating the workload on the partner buffer in experiment 2. With lower production speeds, the average fill grades drop and crane utilisation will be more spread out over the day, introducing more overcapacity to a certain degree, until production is so slow that the lead time is compromised. This is similar in case the crane speed and acceleration are set lower than their maximum, crane utilisation will increase and the overcapacity will be reduced.

Since the exact crane job scheduling, rack location selection and gravity lane selection algorithms are classified, an approximation had to be made on how they work. In reality, these algorithms could work differently yielding different outcomes, although it is expected that this would not cause major differences. This could be confirmed by validating the model with operational data from the real system, which is unavailable as of now since the system is not operational yet. This aspect poses a notable consideration in the research, which may slightly decrease the certainty of the results' validity and therefore it is advised to further validate the model with operational data in the future. Additionally, the conveyor was not modelled in detail. The model only keeps track of the number of pallets waiting on the conveyor for a specific buffer, but it does not take into account that these pallets are also blocking pallets destined for other buffers. Taking this into account might lead to more delays and more pallets waiting on the conveyor.

For each KPI resulting from the simulations, an acceptable margin of error was determined. This should be and was taken into consideration when evaluating the results. In about a quarter of the scenarios, the limit of replications was reached which means that not all KPIs are within the allowed margin of error with a 95% confidence interval. Every time this occurred, the KPI causing this was the aggregate trip loading delay. The variability in this KPI's standard deviation is primarily due to the randomness of the last pallet's placement at the outputs. Depending on the random seed, it may either be placed just in time before downtime begins or delayed by 3 or 8 hours before being placed on the lane. As a result, minor differences in the aggregate trip loading delay were not heavily weighed when evaluating the policies.

The choices made as researcher and modeller also influence the results. A critical research attitude was taken when studying the results of the simulations, no strange behaviour was accepted leading to the discovery of more mistakes which were fixed and which improved the accuracy of results. However, some of the choices made could also negatively impact the accuracy of the results. The choice for the downtime start times and downtime durations might well have influenced the outcome of the results. The downtime durations chosen represent a short downtime and a longer one, with the expectation that at some point, the system will stabilise with a longer downtime and results will not differ with an even longer downtime. But possibly, other policies could perform better with even shorter downtime than 3 hours. However, in reality, it is rarely known in advance exactly how long the downtime will take. Thus, it might be difficult to adjust the policy to execute if it was based on the downtime duration, except for cases of planned maintenance where the total duration is more accurately known in advance. The downtime start times have a larger influence on the results, especially in experiment 2 where both cranes are down, since it dictates the number of already generated pallets for the broken buffer. It was observed that scenario selection influenced the results, leading to differences in performance for varying scenarios. When selecting the downtime start times, worst-case scenarios were chosen so that they cover the peak moments of the day. Similarly, the cranes that break down were chosen to represent a worst-case scenario by choosing the busiest cranes. While this is the safest approach, it might not always be beneficial to pick the best-performing policy based purely on worst-case scenarios which will not always occur. In more relaxed scenarios, it is expected that policies making use of the overcapacity will perform better compared to worst-case scenarios. This would be keeping the capacity at 100% in the experiment with 1 crane down, and rerouting all pallets to the partner buffer in the experiment with both cranes down.

To evaluate the performance of the policies, a scoring system was developed which gives each KPI a score from 1-10 based on a reference minimum and maximum value. Then, the scores are averaged per category and multiplied with weights to come to an overall score per policy. The sensitivity of the results to these weights was proved to be negligible, however, the selection of KPIs for evaluation involves a degree of subjectivity. Additionally, the minimum and maximum reference values are based on the data itself for both experiments combined, but in some cases, a minimum or maximum reference value was chosen manually based on data from scenarios with no downtime and own insight. For example when the values for a certain KPI are very similar for all policies, one does not want to rank them with a score between 1-10 based on minor differences in the values, but rather, give all of them a similar score. To accomplish this, larger reference values were selected, which may contain a minor degree of subjectivity.

10.2. Generalisation

Generalising the results obtained in this research proves difficult since they are highly dependent on the system configuration. The results will most likely be similar for other Order Consolidation Buffers, however, for systems with other configurations and properties, the best-performing policy might be very different. One characteristic that heavily influences the results is the amount of overcapacity in the system at Jumbo. Because of this, in experiment 1, the remaining crane can handle the original workload in most scenarios, and the other buffers do not suffer too much when diverging workload to them. But in a system with less overcapacity, this might cause problems. To remedy this, another policy might have to be developed which unloads the pallets via the failure lane instead of keeping them in the system. However, this entails another characteristic of the system at Jumbo that other systems might not have, which is being able to unload goods through a failure lane and storing them on the floor temporarily. Not having this option also makes policies 1 and 3 impossible in experiment 2. Other examples of characteristics of this system that influence the outcomes and might not be the same in other systems are the order lead times, throughput, production system, ability to load every trip from every dock, and the delivery schedule being known in advance. Similarly, other performance measures might apply in other systems such as the waiting time for a request or the energy consumption.

All things considered, it is evident that the outcomes of this study are particularly relevant to Order Consolidation Buffers, specifically the one implemented at Jumbo Supermarkets. However, through the development of this reusable simulation model, other system configurations can be easily studied.

11 Conclusion

This chapter will conclude the research by answering the sub-questions and main research question, discussing the contribution of this work and giving recommendations for future work. The answers to the sub-questions will be summarised per question:

Sub-question 1: AS/RS research has been going on for a long time, but there are still research gaps such as (partial) downtime and the operation of AS/RS in parallel as seen in OCB.
Sub-question 2: the developed policies vary the workload for a dual-crane AS/RS where one crane broke down from 100% to 0%. If both cranes are down, the policies either store or directly unload the pallets that were already in production in the neighbouring AS/RS, or spread out over the other AS/RS.
Sub-question 3: A model to study downtime in parallel AS/RS should be a DES model to be able to take enough details into account. The model should consist of components modelled as classes of which multiple instances can be created in the simulation such as a pallet, rack, order generator, upstream (production) system, crane and downstream system.
Sub-question 4: Since the model was developed to be a generic model from the start, it contains a lot of options and a large variety of system configurations and scenarios can be studied with the model.
Sub-question 5: If one crane is down, reducing the workload for this AS/RS can minimise performance losses. However, excessively reducing this workload can overload the other AS/RS, compromising system robustness. If both cranes are down, action should be taken to avoid conveyor congestion. The system has enough overcapacity to handle the extra workload originally meant for this AS/RS spread out over the other AS/RS reducing the amount of added manual work.

The main research question: *What is the best operational policy to minimise performance losses while operating parallel AS/RS under partial downtime of the system?* will be answered for the system at Jumbo Supermarkets in two scenarios. In the end, the best-performing policies for both experiments will be compared.

11.1. One of two cranes within an AS/RS down

In this scenario, a decision needs to be made concerning what to do with the incoming workload for this now partially functioning buffer. The 4 developed policies all change the capacity of this buffer with respect to its original capacity. This capacity is varied from the original 100% to 0% where all newly generated throughput avoids this buffer during downtime.

Judging from the results generated by the simulation model, it can be concluded that during an average week, the buffers have enough overcapacity to handle the original workload with just one of the two cranes in operation. However, especially during the busier days of this week, this is less robust and resilient since all of the pallets need to be handled within the reach of the remaining crane, which is ±35% smaller than the full buffer. This means that there are fewer available free locations in the rack and fewer gravity lanes and thus docks can be used. This translates to higher trip frequency per dock and shorter intervals between departure times on a dock, particularly during peak periods of the day.

During a peak day in a peak week, especially with longer downtimes, the remaining crane is not able to handle the full original workload. Operating under the policy retaining the original workload for the partially functioning buffer results in more delays and more congestion on the conveyor network compared to the policies partially shifting the workload away to the other buffers.

A possible downside to shifting workload away to the other buffers is that these buffers might not be able to handle this extra workload causing delays and congestion in those buffers. However, the simulation results indicate that the other buffers have sufficient overcapacity to handle this extra workload, leaving little downsides to shifting workload away to other buffers. The only remaining downside is that shifting too much workload away is less robust and resilient in most cases because of the elevated workload in the other buffers while the buffer with one functioning crane could already handle the workload just fine.

Across all scenarios, the policy reducing the capacity of the buffer with one functioning crane to 50% performs best. Reducing the capacity is not a necessity during quieter days and shorter downtimes, but since there are virtually no downsides to it, it is advised to reduce the capacity of a buffer during partial downtime. Generally, the higher the workload on a day, the more important it becomes to reduce the capacity of a buffer with one functioning crane to minimise delays and upstream system interference and increase robustness and resilience. Considering the current KPIs, there is an optimum for the buffer capacity between 0-100% which would result in the highest score. With 1/2 cranes, 1/2 input points, 8/14 output points and 11/16 rack columns available, it can be argued that this optimum should lie around 50%, which is confirmed by the results. However, the scores for this policy and the policy currently implemented are very close and system adjustments are not essential.

11.2. Both cranes within an AS/RS down

In this scenario, a decision needs to be made about what to do with the pallets destined for this buffer that were already generated and are in production. The 5 developed policies either do nothing as a comparison, unload these pallets via failure lanes or reroute these pallets and give them a destination in another buffer. This is either done at the partner buffer, which is the buffer next to it, or spread out over the other 3 buffers.

Judging from the results generated by the simulation model, it can be concluded that leaving the already generated pallets on the conveyor will cause significant congestion, most likely causing the upstream systems to have to stop production since no more pallets can enter the conveyor network. Additionally, it was observed that the ranking of the policies is similar for each day, and thus, the workload on which they were tested.

Spreading out the workload for both unloading via the failure lane and giving the pallets a destination in another buffer outperforms concentrating this workload on the partner buffer. The advantage of concentrating this workload on the partner buffer would be that pallets stay close to their original destination, but since, as of now, each trip can be loaded from any dock connected to any buffer, this does not bring added value for Jumbo. If in the future it is desired to keep pallets close to their original destination, giving these pallets a destination in the partner buffer is preferred over directly unloading them via the failure lane to prevent added manual work. Delays could occur since the workload becomes too high at peak moments, but on quieter days and with shorter downtimes, the partner buffer can manage.

The total number of pallets that were already generated for a buffer but not produced yet when it breaks down depends on the production pool at that moment. Per trip, this value varies throughout the day and seems to be higher just after production is started, then somewhat steady throughout production, and lower when production is nearing its end. Therefore, the duration of the downtime does not influence the amount of added manual work under policies 1 and 3, but rather the downtime start time. This is apart from some cases where not all already generated pallets are produced within the downtime. In this model, it seems like all pallets that were already generated are produced within 2-4 hours. The downtime duration does influence the performance of the other buffer since during the downtime, newly released trips are spread out over 3 instead of 4 buffers.

Across all scenarios, the policy storing the already generated pallets spread out in the other buffers performs best. This is because the buffers have sufficient overcapacity to handle these extra pallets and unloading them via the failure lanes adds manual work. The policy which unloads the pallets spread out over the other buffers via the failure lanes comes second, outperforming the policy storing all already generated pallets in the partner buffer since it introduces more delays. The policy which unloads all pallets via the failure lane of the partner buffer, which is the policy currently implemented, performs the worst. This is because up to 170 pallets need to exit just one gravity lane in a few hours. This results in a significant amount of added manual work to move the containers fast enough to prevent the failure lane from overflowing and to move these containers to the floor locations which are further away as more containers need to be stored on the floor. A system change is recommended here.

11.3. Experiment comparison

Performances of policies can be compared across both experiments since the same scenarios and reference values were used. This can help in the decision about whether to continue operation with a single crane when one of the cranes breaks down, which slows down repairs, or whether both cranes should be stopped. It can be concluded that across all studied scenarios, it is more beneficial to continue operation with a single crane since all pallets within reach of this crane can still be handled, significantly reducing the amount of delays. However, it should be noted that there are scenarios where stopping both cranes is beneficial, such as with short downtimes or low rack fill grades.

11.4. Contribution

First, the development of this detailed DES model is valuable for other AS/RS researchers. As discussed, there seems to be a lot of duplication of effort with most researchers building their own simulation models for AS/RS. This model will be made publicly available for researchers to reuse and can be applied to a large variety of system configurations while taking the influence of upstream and downstream systems into account.

Similarly, this model is valuable for Jumbo Supermarkets by enabling them to evaluate the performance of their outbound buffer under various circumstances. It was confirmed that their outbound buffer has sufficient capacity to handle their current projected throughput, even under partial downtime of the system. Additional insights were gained while building and using the model which led to a better understanding of the system. In the future, even more insights can be gained by expanding the model and testing the buffer’s performance in different scenarios.

Last but not least, a first contribution was made to research on downtime in AS/RS and how to mitigate its effects, on workload distribution across parallel AS/RS, and on Order Consolidation Buffers, which are areas within AS/RS research which have been neglected in the past. For the system at Jumbo it was determined what works best to mitigate the effects of partial downtime, however, for other systems, other policies might perform better. Therefore, this model can be used to study all sorts of system configurations by adjusting a few parameters in most cases.

11.5. Future work

11.5.1. Research

This work explored the effects of downtime in AS/RS systems and the mitigation of these effects, a topic that has been neglected in AS/RS research. However, there are still ample research opportunities concerning this topic. For example, topics regarding the relation between planned maintenance and resulting downtime, the effects of upstream or downstream system downtime on AS/RS performance or the factors that cause downtime within AS/RS. Additionally, workload distribution across parallel AS/RS, such as Order Consolidation Buffers, can be studied for other system configurations.

A general recommendation that can be made to AS/RS researchers is to focus more on practical research with an impact on industry. Most research is focused on the design or operation of AS/RS in isolation, which is already well-explored. More time should be spent on researching AS/RS in combination with their upstream and downstream systems to further increase their efficiency in practice.

Also, AS/RS researchers could work together more by reducing the duplication of effort with most researchers developing their own models from the ground up. Robust modelling frameworks like the one developed in this work should be shared and further developed for all to use.

11.5.2. Model

Due to the specifics of the system at Jumbo allowing simplifications, time constraints and some algorithms being classified, the model was not developed to its full potential. For future use of the model, improvements can be made to increase the accuracy of the results and broaden the use cases of the model by making the model even more generic. Additionally, a graphical user interface could be developed in order to make it more user-friendly, this was not done now due to time constraints.

Because of the overcapacity of the system at Jumbo, relocations were not taken into account. For other multi-deep systems, there might not be such overcapacity, causing the need for a relocation algorithm. This can be added to the model. Next to this, other assumptions and simplifications could be removed from the model such as the failure lane always being available and the conveyor not being exactly modelled. If the specifics of the algorithms within the system would be shared, the rack location selection, crane job scheduling and gravity lane selection algorithms could also be updated to be closer to reality.

Additionally, the production speed was now modelled as being constant at 199 pallets/h, but in reality, it is more variable. Production speed decreases as the amount of released work to produce drops. Together with the production start and end times, this could be studied more closely and adapted in future versions of the model.

Lastly, more options could be built into the model for future users whose coding skills are limited. The implementation in Python with free libraries allows for easy adjustment to the model to suit each specific system, however, for users with no coding experience, this could be too difficult.

11.5.3. Jumbo

Most importantly, once the system has been operational for a while, Jumbo could collect its operational data and use it for validation of the model. This could further validate the model's representation of the system and could potentially lead to further improvements to the model.

In the future, this model could be applied to other scenarios to study the effects on the performance of their buffers. Examples of topics that can be researched with this model are:

- Effects of relocation of pallets from one buffer to another via the output for empty pallet stacks
- Effects of adjustments to the production schedule and production speeds
- Effects of adjustments to the delivery schedule
- Effects of letting go of the loading order of trips to other distribution centres
- Effects of large amounts of truncations of pallets ('afkappingen')
- Effects of adjusting (physical) system parameters
- Effects of recirculating pallets on conveyor loop to win time instead of unloading through failure lanes
- Finding the downtime duration limit from when stopping both cranes becomes more attractive

For some of these scenarios, the model is already suitable, for others, the model might need adjustments.

Bibliography

Azadeh, K.,, De Koster, R.,, & Roy, D. (2019). Robotized and automated warehouse systems: Review and recent developments. *Transportation Science*, *53*(4), 917–945.

Bansal, V.,, Roy, D.,, & Pazour, J. A. (2021). Performance analysis of batching decisions in waveless order release environments for e-commerce stock-to-picker order fulfillment. *International Transactions in Operational Research*, *28*(4), 1787–1820.

Barbato, M.,, Ceselli, A.,, & Premoli, M. (2019). Evaluating automated storage and retrieval system policies with simulation and optimization. *Advances in Optimization and Decision Science for Society, Services and Enterprises: ODS, Genoa, Italy, September 4-7, 2019*, 127–137.

Bernardo, M., & Hillston, J. (2007). *Formal methods for performance evaluation: 7th international school on formal methods for the design of computer, communication, and software systems, sfm 2007, bertinoro, italy, may 8-june 2, 2007, advanced lectures* (Vol. 4486). Springer.

Bertolini, M.,, Neroni, M.,, & Uckelmann, D. (2023). A survey of literature on automated storage and retrieval systems from 2009 to 2019. *International Journal of Logistics Systems and Management*, *44*(4), 514–552.

Borovinšek, M.,, Ekren, B. Y.,, Burinskienė, A.,, & Lerher, T. (2017). Multi-objective optimisation model of shuttle-based storage and retrieval system. *Transport*, *32*(2), 120–137.

Brailsford, S. C.,, Eldabi, T.,, Kunc, M.,, Mustafee, N.,, & Osorio, A. F. (2019). Hybrid simulation modelling in operational research: A state-of-the-art review. *European Journal of Operational Research*, *278*(3), 721–737.

Chen, W.,, Gong, Y.,, & de Koster, R. B. (2022). Performance estimation of a passing-crane automated storage and retrieval system. *International Journal of Production Research*, *60*(4), 1210–1230.

Chen, Z.,, Li, X.,, & Gupta, J. N. (2015). A bi-directional flow-rack automated storage and retrieval system for unit-load warehouses. *International Journal of Production Research*, *53*(14), 4176–4188.

Ekren, B. Y., & Arslan, B. (2024). A reinforcement learning approach for transaction scheduling in a shuttle-based storage and retrieval system. *International Transactions in Operational Research*, *31*(1), 274–295.

Ekren, B. Y.,, Heragu, S. S.,, Krishnamurthy, A.,, & Malmborg, C. J. (2014). Matrix-geometric solution for semi-open queuing network model of autonomous vehicle storage and retrieval system. *Computers & Industrial Engineering*, *68*, 78–86.

Elsayed, E. A., & Unal, O. (1989). Order batching algorithms and travel-time estimation for automated storage/retrieval systems. *The International Journal of Production Research*, *27*(7), 1097–1114.

Eroglu, E., & Yetkin Ekren, B. (2022). An agent-based simulation model for deadlock prevention in an aisle-to-aisle sbs/rs. *Industrial Engineering in the Internet-of-Things World: Selected Papers from the Virtual Global Joint Conference on Industrial Engineering and Its Application Areas, GJCIE 2020, August 14–15, 2020*, 135–146.

Gagliardi, J.-P.,, Renaud, J.,, & Ruiz, A. (2012a). Models for automated storage and retrieval systems: A literature review. *International Journal of Production Research*, *50*(24), 7110–7125.

Gagliardi, J.-P.,, Renaud, J.,, & Ruiz, A. (2012b). On storage assignment policies for unit-load automated storage and retrieval systems. *International Journal of Production Research*, *50*(3), 879–892.

Gagliardi, J.-P.,, Renaud, J.,, & Ruiz, A. (2014). A simulation modeling framework for multiple-aisle automated storage and retrieval systems. *Journal of Intelligent Manufacturing*, *25*, 193–207.

Gogi, V. S., & Gupta, C. N. (2023). Study & analysis of impact of clustering of machines on performance of automated storage and retrieval system.

Hausman, W. H.,, Schwarz, L. B.,, & Graves, S. C. (1976). Optimal storage assignment in automatic warehousing systems. *Management science*, *22*(6), 629–638.

Hsieh, S., & Tsai, K.-C. (2001). A bom oriented class-based storage assignment in an automated storage/retrieval system. *The international journal of advanced manufacturing technology*, *17*, 683–691.

Jensen, P. A., & Bard, J. F. (2002). *Operations research models and methods*. John Wiley & Sons.

Jumbo Supermarkets. (2022). Jumbo annual report 2022.

Kulturel, S.,, Ozdemirel, N. E.,, Sepil, C.,, & Bozkurt, Z. (1999). Experimental investigation of shared storage assignment policies in automated storage/retrieval systems. *IIE transactions*, *31*, 739–749.

Lee, H. F., & Schaefer, S. K. (1997). Sequencing methods for automated storage and retrieval systems with dedicated storage. *Computers & Industrial Engineering*, *32*(2), 351–362.

Lerher, T. (2017). Design of experiments for identifying the throughput performance of shuttle-based storage and retrieval systems. *Procedia engineering*, *187*, 324–334.

Lerher, T.,, Edl, M.,, & Rosi, B. (2014). Energy efficiency model for the mini-load automated storage and retrieval systems. *The International Journal of Advanced Manufacturing Technology*, *70*, 97–115.

Lerher, T.,, Ekren, B. Y.,, Dukic, G.,, & Rosi, B. (2015). Travel time model for shuttle-based storage and retrieval systems. *The International Journal of Advanced Manufacturing Technology*, *78*, 1705–1725.

Lewczuk, K. (2021). The study on the automated storage and retrieval system dependability. *Eksploatacja i Niezawodność*, *23*(4), 709–718.

Macal, C. M., & North, M. J. (2005). Tutorial on agent-based modeling and simulation. *Proceedings of the Winter Simulation Conference, 2005.*, 14–pp.

Malmborg, C. J. (2001). Rule of thumb heuristics for configuring storage racks in automated storage and retrieval systems design. *International Journal of Production Research*, *39*(3), 511–527.

Man, X.,, Zheng, F.,, Chu, F.,, Liu, M.,, & Xu, Y. (2021). Bi-objective optimization for a two-depot automated storage/retrieval system. *Annals of Operations Research*, *296*(1-2), 243–262.

Marolt, J.,, Kosanić, N.,, & Lerher, T. (2022). Relocation and storage assignment strategy evaluation in a multiple-deep tier captive automated vehicle storage and retrieval system with undetermined retrieval sequence. *The International Journal of Advanced Manufacturing Technology*, 1–18.

Meneghetti, A.,, Dal Borgo, E.,, & Monti, L. (2015). Rack shape and energy efficient operations in automated storage and retrieval systems. *International journal of production research*, *53*(23), 7090–7103.

Meneghetti, A., & Monti, L. (2013). Sustainable storage assignment and dwell-point policies for automated storage and retrieval systems. *Production Planning & Control*, *24*(6), 511–520.

Robinson, S. (2005). Discrete-event simulation: From the pioneers to the present, what next? *Journal of the Operational Research Society*, *56*, 619–629.

Roodbergen, K. J., & Vis, I. F. (2009). A survey of literature on automated storage and retrieval systems. *European journal of operational research*, *194*(2), 343–362.

Rossetti, M. D. (2015). *Simulation modeling and arena*. John Wiley & Sons.

Singbal, V., & Adil, G. K. (2023). Development of an open-source data-driven simulator for the unit-load multi-aisle automated storage and retrieval systems. *Journal of Simulation*, 1–19.

Song, Y., & Mu, H. (2022). Integrated optimization of input/output point assignment and twin stackers scheduling in multi-input/output points automated storage and retrieval system by ant colony algorithm. *Mathematical Problems in Engineering*, *2022*.

Tanaka, S., & Araki, M. (2009). Routing problem under the shared storage policy for unit-load automated storage and retrieval systems with separate input and output points. *International Journal of Production Research*, *47*(9), 2391–2408.

Tappia, E.,, Roy, D.,, Melacini, M.,, & De Koster, R. (2019). Integrated storage-order picking systems: Technology, performance models, and design insights. *European Journal of Operational Research*, *274*(3), 947–965.

Turner, A. (2020). *Evaluation of automated storage and retrieval in a distribution center* [Doctoral dissertation, Massachusetts Institute of Technology].

van der Ham, R. (2018). Salabim: Discrete event simulation and animation in python. *Journal of Open Source Software*, *3*(27), 767.

Wang, J.-Y., & Yih, Y. (1997). Using neural networks to select a control strategy for automated storage and retrieval systems (as/rs). *International Journal of Computer Integrated Manufacturing*, *10*(6), 487–495.

Witt, P. (1974). Automatic storage and retrieval system control. *Proceedings of the May 6-10, 1974, national computer conference and exposition*, 593–611.

Yang, P.,, Miao, L.,, Xue, Z.,, & Qin, L. (2015). Optimal storage rack design for a multi-deep compact as/rs considering the acceleration/deceleration of the storage and retrieval machine. *International Journal of Production Research*, *53*(3), 929–943.

Yu, Y., & De Koster, M. (2009). Designing an optimal turnover-based storage rack for a 3d compact automated storage and retrieval system. *International journal of production research*, *47*(6), 1551–1571.

A

Research paper

Developing a decision support tool for the operation of parallel AS/RS during partial downtime

Luc van den Brink[a], Dik Jansen[b], Frederik Schulte[a], Mark Duinkerken[a]

[a] *Department of Maritime & Transport Technology, Delft University of Technology, Delft, The Netherlands*
[b] *CDC Fresh, Jumbo Supermarkets, Nieuwegein, The Netherlands*

Abstract

This paper investigates the optimisation of Automated Storage and Retrieval Systems (AS/RS) in warehousing by minimising performance losses during partial downtime. Given the increasing automation in logistics, AS/RS systems play a pivotal role, yet the operation of those systems during partial downtime remains a topic ignored in literature. This research fills this gap by exploring the effects of partial downtime in AS/RS through a reusable Discrete Event Simulation model which was developed in Python. This model incorporates the influence of both upstream and downstream systems, a characteristic notably absent from the limited number of publicly-available AS/RS models. Collaborating with Jumbo Supermarkets, the study utilises their highly automated distribution centre with an Order Consolidation Buffer housing 4 dual-crane AS/RS units as a case study. The study identifies operational policies to mitigate partial downtime effects, developed for scenarios with one or both cranes down within an AS/RS. Results suggest strategic workload distribution adjustments among AS/RS can significantly reduce performance degradation, particularly during high workload periods. After comparing both scenarios, it was concluded that for most scenarios, it is beneficial to keep operating the remaining crane when a crane breaks down, even though this slows down repairs. Overall, this research offers insights into parallel AS/RS dynamics under partial downtime and provides practical guidelines for effective operations.

Keywords: Parallel AS/RS, Discrete Event Simulation, Downtime, Modelling framework, Upstream and downstream influence, Order Consolidation Buffer, Workload distribution

---

May 16, 2024

1. Introduction

In the continuously evolving landscape of logistics, warehouses have increasingly embraced automation to cut costs, minimise errors and enhance efficiency. An innovation driving this shift is the Automated Storage and Retrieval System (AS/RS) [1]. AS/RS have been around for a long time but have continuously developed due to technological advances such as robotics, real-time data analytics and machine learning.

In general, AS/RS research is mainly focused on theoretical topics concerning AS/RS in isolation, which has been well-explored but represents a less realistic situation. The influence of upstream and downstream processes, and thus the influence of the total warehouse process, are not considered [2] [3] [4]. This results in a limited impact of AS/RS research on industry. A practical topic within AS/RS research, influenced significantly by upstream and downstream processes, has received minimal attention in the past: their operation under partial downtime. This scenario is applicable in systems featuring multiple AS/RS in parallel and in multi-crane AS/RS configurations.

The goal of this research is to study the influence of partial downtime on the performance of parallel AS/RS, which often exist in Order Consolidation Buffers (OCB), and develop operational policies to mitigate performance losses. This leads to the research question: *What is the best operational policy to minimise performance losses while operating parallel AS/RS under partial downtime of the system?* This question is answered with the development of a reusable Discrete Event Simulation (DES) Model in Python. This model was used to study the Order Consolidation Buffer at the new highly automated distribution centre for fresh products at Jumbo Supermarkets, which will be used as a case study.

The paper is organised as follows: Section 2 gives a brief overview of literature concerning AS/RS. Section 4.4 describes the system at Jumbo Supermarkets which will be used as a case study. Section 4 will describe the methodology used to answer the research question. In section 5, the results will be explained. Finally, sections 6, 7 and 8 discuss the methodology and results, give the conclusion and make recommendations for future research respectively.

2. Literature

AS/RS research has been going on for a long time with initial developments dating back to the 1950s. Since then, a lot of papers concerning the design and operation of these systems have been published. However, there are still plenty of research topics related to AS/RS that are yet to be explored. One topic that has been absent from literature is the operation of AS/RS during partial downtime, which can occur in redundant systems with multiple AS/RS in parallel and in multi-crane AS/RS. This happens to exist in Order Consolidation Buffers, which is another topic that has not been studied in literature, even though they are common in warehouses nowadays [5]. The absence of literature on the aforementioned topics is confirmed by Bertolini et al. [6], who conducted a bibliometric analysis to map the evolution of research themes related to AS/RS by studying over a thousand papers. Both downtime and OCB are keywords that are not present in literature.

Research on the design and operation of AS/RS is often conducted on AS/RS in isolation. Therefore, the influence of upstream and downstream systems on the performance of AS/RS is another topic that has received little attention. Tappia et al. [7] developed an analytical model for an integrated storage-order picking system. This order picking system represents the downstream system of the AS/RS and its influence is thus accurately taken into account. However, the upstream system of the AS/RS is not taken into account.

Gagliardi et al. [8] propose a theoretical model for simulation which can be implemented in a programming language. In this model, there is room to implement a separate supply model upstream of the AS/RS, however, a model to represent downstream processes in the warehouse is not mentioned. Additionally, the model is purely a framework and is not implemented and publicly available.

Next to this, to the best of the author's knowledge, Singbal and Adil [9] are the only ones who developed an open-source DES simulator which can be used to study a large variety of AS/RS configurations. This simulator was developed for multi-aisle AS/RS, which are similar, but still different than parallel AS/RS. It was found that their simulator was limited in the flexibility to be adjusted to accommodate the system under study, for example in the choice of input and output points, the integration of upstream and downstream systems, and more importantly, the integration of downtime.

AS/RS can be modelled in various ways. The most suitable modelling method depends on the research objective:

Queuing Networks (QN) are a stochastic way of modelling in which different variations exist such as Open, Semi-Open and Closed Queuing Networks. A queuing network is a collection of servers, representing the resources of the system, and customers competing for those resources where they possibly have to wait in a queue for those resources. The goal of analysing queuing networks is to determine performance measures such as the number of customers in the system or queue, average time spent in the system or queue and system utilisation factor. The main reason for using QN is the relatively high accuracy and efficient model evaluation [10]. The downsides of QN include the limited degree of detail that can be taken into account.

Mathematical programming (MP) is known as the part of operations research that researches the optimal allocation of resources between competing activities [11]. This way of modelling is deterministic and generally exists of an objective function and a set of constraints with which an optimal solution is desired to be found. The objective function can consist of multiple parts which represent multiple goals, making this type of modelling suitable for multi-objective optimization. Disadvantages of using this method include the scalability problems. When the problem size increases, it can become too computationally demanding. Also, results may be unrealistic when the model assumptions represented in the constraints are too strict.

Discrete Event Simulation (DES) is one of the most popular modelling techniques which has been greatly developed over time [12]. The technique models systems as a sequence of events occurring at discrete moments in time. These events can change the state of the system, or add more events to the events list. Between the events, no changes happen, such that the system jumps in time from event to event until the stop conditions of the simulation have been met.

Agent-Based Modelling (ABM) simulates the interaction and actions of autonomous agents in an environment. It is a way of modelling that has been developed more recently compared to for example DES. This method can be used to study more complex systems with a large number of individual agents by focusing on the individual actions of agents. These models are built bottom-up by identifying agents, defining their behaviour, establishing connections between them and setting environmental variables [13].

3. Problem definition

From the literature review, it can be concluded that topics such as partial downtime in AS/RS and the operation of parallel AS/RS have not gotten any previous attention. This paper will study this topic to provide the industry with guidance on how to deal with this. Partial downtime occurs in redundant systems with multiple AS/RS, possibly with multiple cranes.

To examine a large range of possible solutions to mitigate performance losses in this scenario, the following system characteristics should apply:

1. The inputs of the parallel AS/RS should be connected allowing incoming goods for the upstream systems to be distributed to any AS/RS
2. The system should consist of at least three parallel AS/RS which allows a workload redistribution to either the neighbouring AS/RS or spread out across all other AS/RS
3. The AS/RS should have multiple cranes which allows for the possibility of just one crane breaking down and continuing operation with the remaining crane, possibly with a reduced service area
4. The outputs of the parallel AS/RS should be connected allowing outgoing goods to be distributed from any AS/RS
5. There should be an option to directly unload goods from the AS/RS and remove them from the system which could be necessary in case the extra workload cannot be handled by the other AS/RS

It turns out that systems with these characteristics often exist in the industry in the form of Order Consolidation Buffers. To mitigate performance losses during partial downtime in such a system, guidance on the following decisions is desired:

- Should operation of an AS/RS with one broken crane be continued with the remaining crane, possibly slowing down repairs?
- Should part of the workload for an AS/RSn with one broken crane be redistributed to the other AS/RS?
- How should the workload originally meant for a fully broken AS/RS be redistributed?

4. Methodology

4.1. Policies

The effects of partial downtime were studied in scenarios where one or both cranes within an AS/RS are down. For both scenarios, operational policies which could mitigate performance losses were developed. In the end, a comparison can be made to determine if a single crane should be operated, slowing down repairs, or if it is beneficial to stop both cranes.

In the scenario where one of the cranes is down, the policies vary the throughput capacity of that AS/RS from the original 100% to 0% as seen in table 1. This throughput capacity determines the workload that the AS/RS will receive compared to the other AS/RS.

Policy	**AS/RS capacity**
0	Keep at 100%
1	Lower to 75%
2	Lower to 67%
3	Lower to 50%
4	Lower to 0%

Table 1: Policies when one of the two cranes within an AS/RS is down

In the scenario where both cranes are down, it has to be decided where to send the goods that already started production for that AS/RS. The policies either directly unload these pallets through the neighbouring AS/RS, give them a new destination in the neighbouring AS/RS, directly unload them spread out over the other 3 AS/RS or give them a new destination spread out over the other 3 AS/RS, see table 2.

Policy	**New destination**	**Operation**
0	No change	Wait on conveyor
1	Partner AS/RS	Directly unload at new AS/RS
2	Partner AS/RS	Change destination and keep in new AS/RS
3	Spread over all other AS/RS	Directly unload at new AS/RS
4	Spread over all other AS/RS	Change destination and keep in new AS/RS

Table 2: Policies when both cranes within an AS/RS are down

4.2. Performance evaluation

To evaluate the performance of the different policies, the following performance measures and corresponding KPIs explained in table 3 were selected. Each performance measure was given a weight towards the overall score.

KPI	Weight	Relevant data
Output delay	10	Aggregate trip delay [min]
		Number of trips with delay [trips]
Upstream system interference	8	Maximum amount of pallets on conveyor [pallets]
		Average amount of pallets on conveyor [pallets]
Added manual work	6	Total number of direct unloads [pallets]
		Maximum number of direct unloads for one rack [pallets]
Robustness	4	Maximum crane utilisation [%/h]
		Longest continuous utilisation [min]
		Maximum fill grade [%]
		Minimum margin between departure times per dock [min]
Resilience	2	Maximum average rack fill grade since downtime stop [%pt]
		Maximum average crane utilisation since downtime stop [%]

Table 3: Policy performance evaluation criteria

The **output delay** is considered the most important and is measured by the number of separate trips that arrived at the outputs too late and the total delay of all trips combined.

The **upstream system interference** is considered the second most important performance measure. When too many pallets have to wait at the inputs of the AS/RS, congestion is caused on the connecting conveyor. With too much congestion, the upstream production system might have to come to a standstill since the new pallets cannot be placed onto the conveyor.

Added manual work is created when goods from pallets that are directly unloaded need to be moved to a floor location where they have to be temporarily stored, costing extra money.

Next, **robustness** plays a role towards the overall performance of a policy. This represents the ability of the system to deal with further disturbances and is measured with maximum fill grades, crane utilisation and margin between trip departure times per dock.

Lastly, **resilience** is taken into account, which represents the system's ability to recover from downtime as a result of the different policies. This is measured by the maximum fill grades and crane utilisation since the downtime stopped.

The performances for each KPI resulting from the different policies are scored on a scale from 1-10 compared to a reference minimum and maximum value. These values are based on the best and worst performances across both scenarios, and the performance in a scenario with no downtime.

4.3. Simulation model

The performances of the policies in various scenarios were evaluated through simulation using a DES model. This modelling approach was selected due to the complexity, dynamic nature, and stochastic characteristics of AS/RS, making them challenging to accurately simulate using analytical methods. DES was deemed most suitable for modelling the system as AS/RS allow themselves to be represented as a sequence of discrete events and the amount of separate agents in the system is low. An overview of the model's components and their interaction can be seen in figure 1.

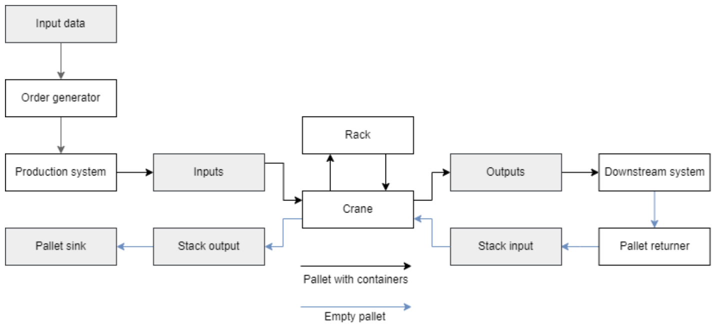


Figure 1: Overview of model structure

The model should be generic with multiple options, allowing it to be used for a large variety of scenarios and system configurations. Additionally, the upstream and downstream systems should be taken into account, it should be possible to introduce downtime and it should be possible to implement several operational policies.

The main assumptions of the model are:

- Trips are spread out equally over the buffers and outputs
- The next pallet from a trip to be produced is completely random
- The next trip from the production pool to produce a pallet for is random with a slight preference for earlier departure times
- Pallets can go directly from production to inputs
- Each trip can be loaded from any dock connected to any buffer
- If one of two cranes is down, the remaining crane will service its full service area
- If it is desired to operate with the remaining crane if a crane is down, added downtime arises from having to move the broken crane to the side
- Relocations do not occur

4.4. Case study

The OCB at a new, highly automated distribution centre of Jumbo Supermarkets in The Netherlands was used as a case study. This OCB consists of 4 dual-crane AS/RS in parallel, making it a suitable candidate for studying the effects of partial downtime. The AS/RS can be operated with 1 crane, but it requires placing the other crane behind a maintenance fence, which incurs additional time. A representation of the overall process in their warehouse can be seen in figure 2.

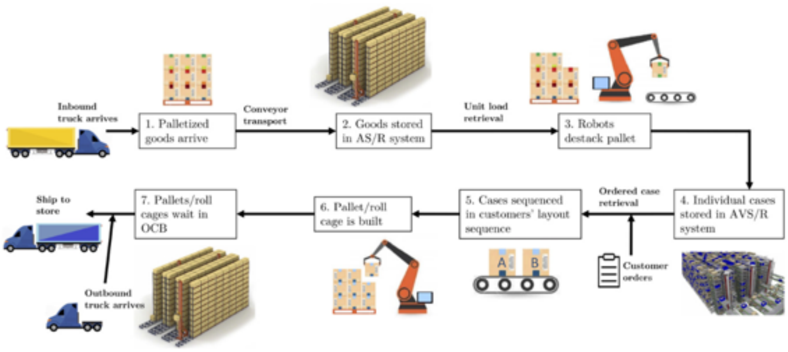


Figure 2: Overview of warehouse system [4]

The upstream system of the AS/RS under study is the production system which processes incoming pallets from suppliers, stores them in AS/RS until needed and destacks the required amount. Then, the individual items are stored in a sequence buffer to make sure that they can be fed into the picking systems in the right order. The picking systems consist of multiple subsystems which place the products onto rollcages manually, partly automated and fully automated. The filled rollcages are placed on pallets in pairs of two and enter the central conveyor. This central conveyor is connected to all AS/RS in the OCB in a loop. The pallets are divided over the AS/RS per trip and once ready to be loaded, move to gravity lanes which are output points that can hold 13 pallets. The pallets must arrive at these outputs in the correct sequence for the rollcages to be loaded into trucks correctly. Additionally, the system is fitted with a failure lane, through which pallets can be unloaded at any time. The empty pallets are returned to the system and placed on the central conveyor. Currently, policy 2 is implemented in the scenario where one crane breaks down and policy 1 is implemented in the scenario where a whole AS/RS breaks down.

4.5. Model verification and validation

The model underwent rigorous verification to ensure it behaves as intended. This included balance checks to confirm consistency between produced and loaded pallets, seed independence tests with varying random seeds, and consistency checks on hypotheses. Hand calculations, deterministic runs, and unit tests further verified the model's behaviour.

Stochastic distributions were verified by comparing simulated times with actual event log data. Plots were used to verify crane tasks, rack fill grades, and overall system behaviour, ensuring alignment with expected values.

Sensitivity analyses were conducted on estimated input parameters to assess their impact on simulation results, most parameters had little influence, and the ones with a larger influence were re-evaluated. Overall, the verification process confirmed that the model behaves as intended.

Since the system at Jumbo is not operational yet, the model could not be validated with real-world data. Experts on the system were consulted who confirmed that the model represents the system's design and operation.

Additionally, an analytical model within Jumbo was used to predict production and fill grades, allowing comparison with simulation results. Although slight discrepancies exist due to differences in assumptions, overall trends aligned, validating the simulation's logic.

Parameter estimations, such as crane travel times and pallet pickup times, were validated using data from similar systems and design figures. Calculations based on Witron's design figures matched expected values, further validating the model.

4.6. Experimental plan

The performance of the system under the policies was evaluated during varying conditions. Three input datasets were used that represent an average day in an average week, a peak day in an average week and a peak day in a peak week. These datasets determine the moment production is released, the number of roll cages per client and thus the loading sequence, and the departure time for each trip. Furthermore, for each dataset, simulations were run with a start time of the downtime at 05:00, 07:00 and 13:00 and a downtime duration of either 3 or 8 hours. These parameters were chosen to cover the peak moments of the day and to cover moments when both production was already started, or there was no production at the downtime start time.

As part of the experimental plan, the simulation parameters including initialisation, run length, and the number of replications were determined.

To create a realistic scenario, the system is partially filled at the start of the simulation based on the production day before. Trips released until the end of the target day are included to ensure continuous system operation and accurate representation of fill grades. In figure 3, the warm-up period of the simulation for Friday as the target day and thus including Thursday can be seen.

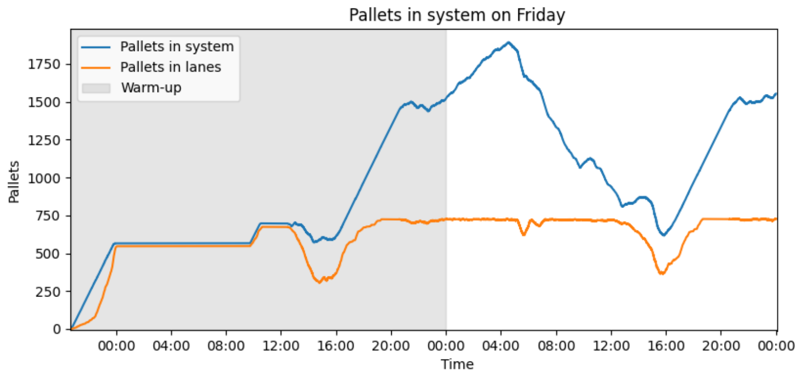


Figure 3: Warm-up period of simulation on Friday

The simulation examines KPIs over a full production day, as daily operations follow similar patterns. Due to relatively fast system recovery times, additional simulation long after downtime stopped is unnecessary.

Multiple simulation runs with varying random seeds ensure robust results. The required number of replications is calculated based on desired confidence levels and allowed margins of error for each KPI. Typically, 50-100 replications were sufficient, in some cases the maximum number of replications of 200 was reached.

To expedite simulations, profiling identified resource-intensive code sections, while parallel processing maximized CPU usage and thus minimised computation time. However, attempts with Just-In-Time compilation did not yield efficiency gains due to Python's inherent efficiency in newer versions. Running 8 simulations in parallel proved optimal for the used computer, with each instance executing with a unique random seed for comprehensive result collection.

5. Results

The simulation experiments delved into strategies for downtime effect mitigation of AS/RS cranes in scenarios where one or both cranes within an AS/RS experience downtime. This is an overview of the results:

5.1. One of two cranes within an AS/RS down

Average day in an average week

Throughout the day, all policies performed similarly due to the low throughput. No issues regarding the amount of pallets on the conveyor were observed. Although the maximum system fill grade rose slightly due to the broken crane, it remained moderate at $\pm 50\%$. When reducing the capacity of the partially broken AS/RS to 0% under policy 4, the other AS/RS managed the extra workload but with slightly reduced robustness. Notably, policy 0, 100% capacity, reduced the minimum margin between departure times per dock in some scenarios, indicating slightly reduced robustness.

Examining individual results, shifting the workload to the other functional buffers aided broken crane recovery. Policies retaining a higher workload caused more added manual work since more pallets arrive at the buffer while moving the repaired crane back into action.

Peak day in an average week

Despite handling higher volumes, the remaining crane coped well with the full workload, with no significant delays or conveyor queues observed. As expected, crane utilisation and fill grades were higher compared to an average day but did not pose any issues.

Notably, in the scenario with downtime from 05:00 to 14:15, the minimum margin between departure times on a dock decreased to just 1 hour in the broken buffer under policies 0 and 1, which could be problematic for truck loading schedules. Under policy 4, 0% capacity, this minimum margin decreased to 90 minutes in the other buffers due to their increased workload.

Under policy 0, 100% capacity, the remaining crane's utilisation during downtime was significantly higher compared to policy 3, 50% capacity. Shifting the workload away from the partially broken AS/RS improved its robustness, but excessive workload shifts, as seen in policy 4, 0% capacity, reduced the robustness of the other AS/RS.

Overall, the system's performance on this day was consistent across policies, with minor variations in crane utilisation and departure time margins, highlighting the importance of workload distribution for system resilience.

Peak day in a peak week

During this day, the policies exhibit significant performance variations, particularly during downtime from 05:00 to 14:15. Policy 0, 100% capacity, shows considerable delays, slightly more congestion on the conveyor, and a continuous crane utilisation exceeding 4 hours. This strain on the remaining crane after downtime affects the margin between departure times, with the minimum margin being just 30 minutes, causing potential loading issues for successive truck drivers. An overview of the pallets in different stages throughout the simulation can be seen in figure 4.

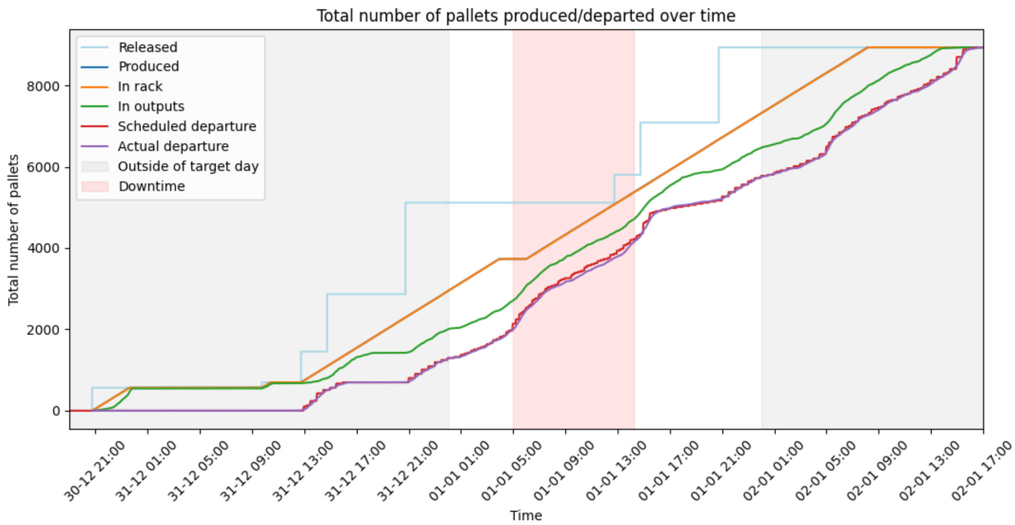


Figure 4: System overview with downtime from 05:00 - 14:15

In contrast, policy 3, 50% capacity, performs notably better in this scenario, showcasing improved system efficiency and departure scheduling. The overall performance differences across scenarios are evident in the generated scores, with policy 0, 100% capacity, consistently ranking lower due to increased delays and interference with upstream systems.

Policy 0, 100% capacity, causes more delays, upstream system interference, and lower resilience. The tightest departure time margin occurs during downtime from 05:00 to 14:15, reflecting the peak production period.

The maximum system fill grade is similar across scenarios since it is reached before downtime starts, except for policy 4, 0% capacity, where the maximum fill grade is reached later due to the severely altered workload distribution.

Overall scores

The combined scores across all scenarios can be found in table 4. This underscores that the policy maintaining the workload for the partially broken buffer performs the worst, though still within acceptable bounds across most scenarios. Policy 3, 50% capacity, emerges as the most effective, striking a balance between redistributing workload to other buffers and retaining an appropriate workload in the buffer with one operational crane. However, policy 2, which is currently implemented, has a similar score and performs almost as well.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	9,6	9,7	9,9	9,8	9,7
Upstream system interference	8	9,2	9,6	9,6	9,8	9,9
Added manual work	6	3,5	3,8	3,9	4,2	5,1
Robustness	4	7,4	8,3	8,7	9,2	8,1
Resilience	2	9,7	9,7	9,6	9,1	7,7
Overall		**8,0**	**8,3**	**8,5**	**8,6**	**8,5**

Table 4: Combined scores experiment 1

5.2. Both cranes within an AS/RS down

Average day in an average week

Policy 0, no action, consistently leads to congestion, with over 100 pallets accumulating on the conveyor, prompting upstream system shutdowns. Interestingly, the maximum pallet count on the conveyor remains unaffected by downtime duration, as the buffer's capacity drops to zero when downtime starts, diverting new workload to other buffers.

Comparing policies 1 and 3, both unloading pallets via failure lanes, differences primarily emerge in maximum failure lane unloads for a single buffer and crane utilisation. Additionally, policy 1, unload at partner, exhibits higher crane utilisation due to concentrated unloading in one buffer, needing rapid movement of pallets to the failure lane.

Conversely, policies 2 and 4, which reroute pallets to alternative buffers, exhibit variations in maximum crane utilisation, departure time margins, and post-downtime fill grades. All policies involve failure lane unloads since the pallets from rerouted trips which are stuck in the broken buffer are unloaded at the failure lane afterwards, though policies 1 and 3 require more manual work. However, once this manual workload is cleared under the policies unloading all pallets, buffer recovery is smoother.

Peak day in an average week

Policies 2 and 4 exhibit reduced failure lane unloads compared to the average day, despite higher overall throughput. This is attributed to the increased trip frequency per dock. Because of this, trips move to the gravity lanes later since they are available later, allowing more trips from the broken buffer to still be rerouted. However, under policy 2, store in partner, a notable concern arises with a minimum margin of just 33 minutes between departure times when downtime commences at 07:00. Additionally, this policy leads to elevated crane utilisations, system fill grades, and aggregate delays.

During a 3-hour downtime, not all pallets intended for the broken buffer are produced yet, resulting in slightly fewer pallets on the conveyor compared to an 8-hour downtime under policy 0, no action. Despite similar upstream system interference across policies, policies 1 and 3 demonstrate greater robustness in rack usage but less so in crane utilisation. This discrepancy arises from the need for crane movement in transferring pallets to the failure lane, impacting maximum crane utilisation.

Peak day in a peak week

Delays are marginally higher under policy 2, store in partner, but differences across policies are negligible. The workload distribution does not significantly impact the aggregate delay, as pallets stuck in the broken buffer dictate overall delays. Despite concerns about crane workload, even policy 2, which concentrates workload on one buffer, shows no significant peak in pallets waiting on the conveyor since the downtime began.

In an average week, a production dip before 13:00 in combination with new trips being released at 12:45 makes sure that at 13:00, the production pool consists mostly of new trips of which few pallets were already produced. On this day, there is no production dip, which results in fewer pallets already being generated but not produced yet. This leads to fewer pallets to reschedule compared to the peak day in an average week.

Under policy 2, store in partner, crane utilisation exceeds 90% since downtime stopped, posing a robustness issue. Spreading workload across buffers proves more resilient, although potential downsides exist for later replanning of the rerouted rollcages. Overall, distributing workload over multiple buffers performs better, especially if pallets are given a destination in the other buffer, reducing manual work.

Overall scores

The combined scores across all scenarios can be found in table 5. The performance of the policies with respect to each other does not differ much per day. The policies spreading out the workload always perform better than the policies concentrating the workload on the partner buffer. Moreover, the policies giving the pallets a new destination always perform better than the policies directly unloading them since manual work is saved and there is sufficient overcapacity for the other buffers to handle the extra workload. It is advised to change the currently implemented policy to policy 4.

		Policy				
Scores	**Weight**	**0**	**1**	**2**	**3**	**4**
Output delay	10	4,9	6,7	6,5	6,8	6,7
Upstream system interference	8	1,0	9,4	9,8	9,9	9,9
Added manual work	6	10,0	1,0	7,4	3,7	7,4
Robustness	4	7,2	6,5	5,2	8,1	7,7
Resilience	2	7,5	7,4	5,4	7,5	6,9
Overall		**5,4**	**6,3**	**7,3**	**7,2**	**7,8**

Table 5: Combined scores experiment 2

5.3. Experiment comparison

A comparison of performance metrics across both experiments can be seen in table 6. From the combination of the metrics, and the scores for the best-performing policies specified in earlier tables, it can be seen that it is beneficial to continue operation of a single crane for the scenarios studied, even though repairs are slowed down.

		Output delay		**Upstream system interference**		**Added manual work**		**Robustness**				**Resilience**	
	Remaining crane	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
Day 1	Operate	1312,3	41,4	0,03	5,16	66,9	39,3	90,2	54,2	50,0	151,0	36,8	43,0
	Stop	1374,2	42,6	0,03	5,54	42,6	32,7	93,6	63,7	53,4	150,7	40,6	46,0
Day 2	Operate	1462,3	57,5	0,02	3,93	83,2	46,8	92,6	61,6	54,8	120,5	39,8	47,7
	Stop	2055,9	58,7	0,01	3,61	31,5	17,1	93,8	62,8	58,0	99,5	44,6	51,5
Day 3	Operate	1902,9	57,6	0,04	4,54	77,9	42,8	99,4	108,3	57,7	88,5	40,0	52,7
	Stop	2682,4	62,0	0,02	4,03	20,8	15,0	98,0	110,5	58,6	73,2	45,3	60,0

Table 6: Performance metric comparison of best-performing policies for both experiments

6. Discussion

The simulation model developed for Jumbo's system provides a comprehensive analysis, yet it relies on assumptions and simplifications, potentially impacting the results. Sensitivity analyses revealed minor variations in results when altering most approximated values, but significant deviations in production speed can substantially alter outcomes. Similarly, if the crane speed and acceleration are set lower than their maximum, the cranes will have less overcapacity.

The accuracy of the model depends on approximations of algorithms for crane job scheduling, rack location selection and output selection, which may differ in practice. The accuracy of the model could be confirmed by validation with operational data, which is unavailable. This diminishes the certainty of the result and underscores the importance of future validation endeavours.

The acceptable margin of error was determined for each KPI. However, reaching the replication limit in some scenarios could affect the determined values for specific KPIs, such as the aggregate trip loading delay. The researcher's decisions, such as downtime start times and durations, also wield influence over outcomes.

It was noticed performances differ during varying scenarios. Often, worst-case scenarios, such as the busiest cranes and buffers breaking down, were chosen which may potentially create a misleading impression. Relaxed scenarios might favour policies relying more on the system's overcapacity.

Additionally, scoring the policies involves subjectivity in selecting KPIs, underscoring the need for careful consideration. Furthermore, the generalisability of results is limited to systems with similar configurations, such as Order Consolidation buffers. This is because outcomes are significantly influenced by factors such as the AS/RS characteristics like overcapacity, order lead times, ability to unload through a failure lane, and characteristics of the upstream and downstream systems.

While the study's conclusions offer valuable insights into Jumbo's Order Consolidation Buffer, the development of the reusable simulation model enables the examination of different system configurations in future research.

7. Conclusion

The main research question: *What is the best operational policy to minimize performance losses while operating parallel AS/RS under partial downtime of the system?* can be answered for both studied scenarios.

For the case study at Jumbo Supermarkets, the two scenarios considered are scenarios where one or both of the cranes within an AS/RS are down. To answer this question, operational policies were developed for both scenarios. KPIs were defined, and a scoring system was created to evaluate buffer performance. Using a detailed reusable DES model developed in Python, the system's performance under various policies, downtime start times, and durations were examined across three days with varying workloads.

In the scenario with one crane down, reducing the capacity of the partially functioning buffer proved to be the most effective approach. Even though the partially functioning buffer could manage the original workload in most scenarios, reducing the capacity helped minimise delays and congestion, enhancing system robustness and resilience. Across all scenarios, reducing the buffer's capacity to 50% performed best, optimising between workload reduction for the remaining crane and workload increase for the other buffers.

In the scenario with both cranes down, policies that spread out workload, either through unloading via failure lanes or rerouting pallets to other buffers, outperformed policies concentrating workload on a single buffer. Giving pallets a destination spread out over other buffers yielded the best results, leveraging system overcapacity and minimising added manual work.

Comparing policy performances across experiments, using identical scenarios and reference values, aids in deciding whether to maintain single-crane operation during breakdowns, slowing repairs, or stop both cranes. Overall, operating with a single crane proves advantageous across most scenarios, as it can handle all reachable pallets, notably minimising delays. However, with very short downtimes, stopping both cranes yields the best performance.

Although these results might be most relevant to Order Consolidation Buffers and specifically the one at Jumbo Supermarkets, the developed DES model can be used to study the results for other systems.

8. Recommendation

This study delved into the impact of downtime in AS/RS systems and strategies for mitigation, a topic often overlooked in AS/RS research. While valuable insights were gained, numerous research topics remain unexplored. For instance, future research could investigate the relation between planned maintenance and downtime, the repercussions of upstream or downstream system downtime on AS/RS performance, or the root causes of downtime within AS/RS. Additionally, workload distribution across parallel AS/RS can be further studied in other scenarios.

A broader recommendation for AS/RS researchers is to focus more on practical research with tangible industry impact. Rather than solely concentrating on AS/RS design in isolation, future efforts should explore the operation of AS/RS in conjunction with their upstream and downstream systems to enhance real-world efficiency. Moreover, collaboration among AS/RS researchers to share robust modelling frameworks could streamline efforts and foster innovation.

Due to time constraints and the classification of algorithms, the model in this study was not fully optimised. To enhance its accuracy and versatility, several improvements could be implemented. For instance, incorporating a graphical user interface would enhance usability. Additionally, refining the model to account for factors such as relocations, variable production speeds, and an exact conveyor representation would yield more precise results.

Once the Jumbo system has been operational for a while, validating the model with operational data could enhance its accuracy and applicability. Additionally, the model could be utilised to explore various scenarios affecting buffer performance, including pallet relocation among buffers, production schedule adjustments, and delivery schedule modifications. These applications could provide additional valuable insights into system optimisation and inform decision-making processes.

In summary, future work should focus on refining the model, validating it with operational data, and exploring its applications in various scenarios to further advance the understanding and practical implementation of AS/RS systems.

References

[1] H. F. Lee, Performance analysis for automated storage and retrieval systems, IIE transactions 29 (1) (1997) 15–28.

[2] K. J. Roodbergen, I. F. Vis, A survey of literature on automated storage and retrieval systems, European journal of operational research 194 (2) (2009) 343–362.

[3] J.-P. Gagliardi, J. Renaud, A. Ruiz, Models for automated storage and retrieval systems: a literature review, International Journal of Production Research 50 (24) (2012) 7110–7125.

[4] K. Azadeh, R. De Koster, D. Roy, Robotized and automated warehouse systems: Review and recent developments, Transportation Science 53 (4) (2019) 917–945.

[5] K. Azadeh, R. De Koster, D. Roy, Robotized warehouse systems: Developments and research opportunities, ERIM report series research in management Erasmus Research Institute of Management (ERS-2017-009-LIS) (2017).

[6] M. Bertolini, M. Neroni, D. Uckelmann, A survey of literature on automated storage and retrieval systems from 2009 to 2019, International Journal of Logistics Systems and Management 44 (4) (2023) 514–552.

[7] E. Tappia, D. Roy, M. Melacini, R. De Koster, Integrated storage-order picking systems: Technology, performance models, and design insights, European Journal of Operational Research 274 (3) (2019) 947–965.

[8] J.-P. Gagliardi, J. Renaud, A. Ruiz, A simulation modeling framework for multiple-aisle automated storage and retrieval systems, Journal of Intelligent Manufacturing 25 (2014) 193–207.

[9] V. Singbal, G. K. Adil, Development of an open-source data-driven simulator for the unit-load multi-aisle automated storage and retrieval systems, Journal of Simulation (2023) 1–19.

[10] M. Bernardo, J. Hillston, Formal Methods for Performance Evaluation: 7th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2007, Bertinoro, Italy, May 8-June 2, 2007, Advanced Lectures, Vol. 4486, Springer, 2007.

[11] P. A. Jensen, J. F. Bard, Operations research models and methods, John Wiley & Sons, 2002.

[12] S. Robinson, Discrete-event simulation: from the pioneers to the present, what next?, Journal of the Operational Research Society 56 (2005) 619–629.

[13] C. M. Macal, M. J. North, Tutorial on agent-based modeling and simulation, in: Proceedings of the Winter Simulation Conference, 2005., IEEE, 2005, pp. 14–pp.

B

Pseudo code

B.1. Input data processing

for each row in the dataset
    if this is the first row of a new trip
        >append the collected data for this trip to the new dataset
    if mix is true
        >set containers to the value in the row - 1
    else
        >set containers to the value in the row
    if the amount of containers in this row is uneven
        >set pallets to the upwards rounded value of containers / 2
        >set mix to true
    else
        >set pallets to containers / 2
        >set mix to false

B.2. Pallet Generator

B.2.1. Functions

function next_trip_to_produce
    while the production pool is not full
        >filter dataset to only include trips which are released now
        if there are no new trips released
            hold until the next trip is released
        if next trip has departure time after production stop this day *and* now is before production restart
            break
        >add the trip with the earliest departure time to the pool of trips to be generated
        >remove this trip from the list of trips to be produced

function determine_next_rack
    while true
        >set the next rack from the order
        >set the capacity of the next rack
        if the rack capacity is 1
            return this rack
        else if the rack capacity is 0
            continue
        else
            if this rack needs to be skipped because of its capacity
                continue

```
    else
      return this rack

function determine_lanes
  if one crane is down
    >determine the available gravity lanes based on service area of remaining crane
  else
    >set all gravity lanes as available
  >find the available gravity lanes which will be empty first
  return these gravity lanes
```

B.2.2. Process

```
while true
  if there are still trips to be produced
    call next_trip_to_produce
  else
    passivate
  while there are trips in the pool to be generated
    >select the first trip from the pool
    if there is a wait time for this trip to start production
      hold for this time
    call determine_next_rack
    call determine_lanes
    >determine the input point in the rack based on the determined gravity lanes
    >create the new pallet with all determined properties
    >reduce the number of pallets to be produced for this trip and sequence by 1
    >increase the sequence if all pallets of a sequence were generated
    >add this pallet to the production queue
    if the pallet maker is passive
      activate pallet maker
    if all pallets of this trip were generated
      >remove this trip from the pool of trips to be generated
      >shuffle the order of the pallets of this trip in the production queue
      call next_trip_to_produce
  passivate
```

B.3. Pallet Maker

B.3.1. Functions

```
function send_to_failure_lane(pallet)
  >set the status of the pallet to 'Failure'
  >set the current coordinates of the pallet to those of the input point
  >set the destination of the pallet to those of the failure lane
```

B.3.2. Process

```
while true
  if there are no trips in the production pool
    passivate
  >determine the odds of being produced for every trip in the production pool
  >randomly choose a trip from which a pallet will be produced based on the odds
  if experiment is 1
    if both cranes are currently out of operation
      >send this pallet to the failure lane of the partner buffer
    else if the crane supposed to handle this pallet is down
      >change the input point to the one of the other crane
      if any of the pallets belonging to this trip is blocked by the broken crane
```

```
            call send_to_failure_lane(pallet)
        else
            >set the current coordinates of the pallet to those of the new input point
  else if experiment is 2 and policy is not 0 and the cranes are down
      if policy is 1 or 3 or any of the pallets of this trip are already in the broken rack
          if policy is not 3
              >switch pallets of this trip to partner rack
          else
              >switch this pallets of this trip to another rack, spread out
          >change the input point of this pallet
          call send_to_failure_lane(pallet)
      if policy is 2 or 4
          if policy is 2
              >switch pallets of this trip to partner rack
          else if policy is 4
              >switch pallets of this trip to another rack, spread out
          for each gravity lane combination of the new rack
              if the last trip going to these lanes does not depart later than this one
                  >determine the difference in departure times with the last trip
              else
                  >determine the smallest difference in departure times with the last and second to last trip
          >set the lanes of this trip to the ones with the largest minimum difference in departure times
          >determine the input point of this pallet
          >set the current coordinates of this pallet to those of the input point
  hold for the production time sampled from a normal distribution
  if there is space in the targeted input point queue of the rack
      >place this pallet into that input queue
  else
      >place this pallet onto the conveyor
  if the targeted crane is passive
      activate that crane
  if this pallet is currently the only one in the input point queue
      >add this pallet to the crane task queue
  >remove this pallet from the production queue
  if this was the last pallet of the trip to be produced
      remove this trip from the production pool
      if the pallet generator is passive
          activate pallet generator
```

B.4. Crane

B.4.1. Functions

```
function find_three_deep_location(pallet)
  if experiment is 1
      >determine if a crane is down and which one
      if crane 0 is down
          >set the locations to choose from to the free 3-deep locations in reach of crane 1
      else if crane 1 is down
          >set the locations to choose from to the free 3-deep locations in reach of crane 0
      else
          >set the locations to choose from to all free 3-deep locations
  else
      >set the locations to choose from to all free 3-deep locations
  if there are no free 3-deep locations
      >stop the simulation
  if there are free locations above the gravity lanes of the pallet
```

```
    return the lowest location above these gravity lanes
  else
    return the location that results in the smallest travel distance from input to rack location to output

function find_two_or_one_deep_location(pallet, previous pallet)
  if previous pallet is currently in operation
    return the location of the previous pallet
  else if previous pallet is not in the rack yet
    >set the previous location to the destination of the previous pallet
  else
    >set the previous location to the current location of the previous pallet
  return the location in front of the previous location

function ready_for_lane(pallet)
  >find the previous trip scheduled for the same outputs as this pallet
  if any of the pallets of the previous trip is not at the output yet
    return false
  if pallet belongs to first client of the trip or all pallets of previous clients are already at the output
    if there are less pallets at the first output than the second and this output is not full
      return true and output 0
    else if there are less pallets at the second output than the first and this output is not full
      return true and output 1
    else
      return false
  else if there is just 1 pallet of the previous sequence not at the output yet
    if both outputs now equally full and not more than half of the pallets for this client at output
      if output 1 is not full
        return true and output 1
      else
        return false
    else if the outputs are not equally full and not more than half of the pallets for this client at output
      if fullest output is not full yet
        return true and the fullest output
      else
        return false
    else
      return false
  else
    return false

function space_in_glanes(pallet)
  >determine the number of pallets on their way to the gravity lanes of this pallet
  >determine the number of pallets in each gravity lane of this pallet
  if the total number of pallets in and planned for the gravity lanes is less than the capacity
    return true
  else
    return false

function space_in_front(pallet)
  if the location in front of this pallet is free
    return true
  else
    return false

function no_pal_of_sequence_coming(pallet)
  >determine the number of pallets for this client on the conveyor
```

```
    >determine the number of pallets for this client at the inputs
    >determine the number of pallets for this client in the production queue
    if the total of these numbers is 0
        return true
    else
        return false

function determine_location(pallet, check)
    if there are no free locations
        >stop the simulation
    if space_in_lanes and ready_for_lane
        >set the determined location to the gravity lane returned by ready_for_lane
    else
        if there is a pallet of this trip in the rack with the same sequence and space_in_front
            >set the determined location to the location returned by find_two_or_one_deep_location
        else if pallet of this trip exists in rack with later sequence and space_in_front and no_pal_of_sequence_coming
            >set the determined location to the location returned by find_two_or_one_deep_location
        else
            >set the determined location to the location returned by find_three_deep_location
    if check is true
        return the determined location
    >set the location of the pallet to the determined location
    if the determined location is in the rack
        >make the determined location occupied

function find_earliest_release_time
    for each pallet in the crane queue
        if the release time of this pallet is earlier than the last stored earliest release time
            >set the earliest release time to the release time of this pallet
    return the earliest release time

function calc_moving_time(x1, y1, z1, x2, y2, z2)
    >calculate the horizontal distance in meters based on the coordinates
    >calculate the vertical distance in meters based on the coordinates
    if the crane cannot reach the maximum speed on the horizontal distance
        >calculate the horizontal travel time with formula specified in section 5.4.5
    else
        >calculate the horizontal travel time with formula specified in section 5.4.5
    if the crane cannot reach the maximum speed on the vertical distance
        >calculate the vertical travel time with formula specified in section 5.4.5
    else
        >calculate the vertical travel time with formula specified in section 5.4.5
    return the maximum of either the horizontal or vertical travel time

function task_possible
    if the pallet needs to go to the failure lane
        return true
    if it is not past the release time of the pallet
        return false
    if pallet is headed to gravity lane
        if these gravity lanes are full
            return false
        if not ready_for_lane
            return false
    if the other crane is currently operating around the same location of this task
        return false
```

```
    return true

function trajectory_collision(pallet)
    >set the moving time to the time returned by calc_moving_time
    if pallet does not go to failure lane and is not an empty stack
        >set determined location to location returned by determine_location
    >calculate the time it would take to move this pallet with calc_moving_time
    >set the trajectory this crane would have when performing this task
    >compare trajectories of both cranes and determine if at any moment they will be too close
    if the cranes would be too close
        return true
    else
        return false

function crane_obj_function(pallet)
    if not task_possible
        >set score to -1000000
    else
        >set score to current timestep in seconds - departure time in total seconds - pallet sequence
        if pallet is at input point
            if there are 2 pallets at the input point
                >set score to -1800
            else if there are more than 2 pallets at the input point
                >set score to 7200
        else if the pallet is an empty stack at its input point
            >set score to 7200
        if trajectory_collision
            >reduce the score by 200000
    return the score

function check_collision_range
    >determine the exact current location of the other crane
    if the cranes are too close
        if one of the cranes is passive
            >signal that the cranes are too close while one is idle
        else
            >signal that the cranes are too close
    else if the crane is outside of its operating range
        >signal that the crane is outside of its operating range

function check_idle_crane_collision
    if the other crane is passive and it is in the way of the task of this crane
        >move the passive crane out of the way
        return true
    else
        return false

function move_to_idle_position
    >move this crane to its dwell point
```

B.4.2. Process

```
while true
    while true
        if the downtime of this crane started
            hold until end of downtime
        if there are no pallets in the crane queue
```

```
   call move_to_idle_position
   passivate
   break
>determine the scores of all tasks in the crane queue with crane_obj_function
>sort the crane queue with scores from high to low
>set the task with the highest score
if the tasks with the highest score is possible
   if there are more tasks with this highest score
      >choose the task which starts closest to the location of the crane
   else
      >choose the single task with the highest score
   if the pallet from this task is in the rack
      >make this location available
   else if the pallet needs to go to the failure lane
      >set the destination of the pallet to the failure lane
   else if this is not an empty pallet stack
      call determine_location
   else if this is an empty pallet stack
      >remove this stack from the other crane queue
      >set the gravity lane to the one determined by ready_for_lane
   if performing the task would cause a collision with the other crane
      >move this crane out of the way
      hold until the end of the task of the other crane
else
   if the crane needs to wait for a pallet to be released
      call move_to_idle_position
      activate at earliest release time
   else
      call move_to_idle_position
      passivate
   break
call check_idle_crane_collision
>calculate the time it takes to move to the pallet
hold for the moving time
>update the crane location
call check_collision_range
>calculate the time it takes to move the pallet to its destination
hold for the operation time
>remove the task from the crane queue
if pallet was at an input
   >remove pallet from input queue
   if there are more pallets at the input
      >move the first pallet at the input into the crane queue
      if there are pallets waiting on the conveyor
         >move this pallet to the input
if it is an empty pallet stack
   >remove stack from empty stack input
   if there are more empty stacks waiting
      >move this stack into the crane queue
if pallet needs to go to gravity lane
   if pallet comes from the rack
      >remove it from the rack
      if there is a pallet in the rack behind this one
         >move this pallet to the crane queue
   >add the pallet to the gravity lane
   if the loader of that gravity lane is passive
```

```
        activate that loader
    if the pallet needs to go to the failure lane
      >change pallet status to loaded
    else if pallet needs to go to the rack
      >move pallet into the rack
    >change the location of the pallet and crane to the destination
    call check_collision_range
    if the pallet is now placed in the rack
      >change the destination of the pallet to its gravity lanes
      if pallet was placed in front of another one
        >remove that one from the crane queue
      >move this pallet back into the crane queue
```

B.5. Loader

B.5.1. Process

```
while true
  while there are pallets in the lanes of this loader
    if both lanes have pallets in them
      if the first pallet in the first lane has the earliest departure time
        >set the lane to load from next as the first lane
      else if the first pallet in the second lane has the earliest departure time
        >set the lane to load from next as the second lane
      else
        if the first pallet in the first lane has a lower sequence
          >set the lane to load from next as the first lane
        else if the first pallet in the second lane has a lower sequence
          >set the lane to load from next as the second lane
        else
          >set the lane with the most pallet from this sequence remaining as lane to load from next
    else if just one lane has pallets in them
      >set this lane as the lane to load from next
    if the first pallet of the lane to load from is not ready to be loaded yet
      hold until the release time of this pallet
      continue
    hold for the sampled loading time of this pallet
    >remove this pallet from the gravity lane
    >move this pallet to the empty pallet queue
    if the pallet returner is passive
      activate the pallet returner
    if the crane servicing this lane is passive
      activate that crane
```

B.6. Pallet returner

B.6.1. Functions

```
function empties_to_crane_queue
  if both cranes are passive
    >send the pallet to the closest crane queue
  else if the first crane is passive
    >send the pallet to the first crane queue
  else if the second crane is passive
    >send the pallet to the second crane queue
  else if both cranes are active
    >send the pallet to both crane queues
```

B.6.2. Process

while true
    if there are more pallets in the empty pallet queue than needed to form a stack
        >remove these pallets from the empty pallet queue
        >create the empty pallet stack
        >send this stack to the empty pallet stack input
        call *empties_to_crane_queue*
    passivate

B.7. Results generator

B.7.1. Functions

function *run_script*(first_seed, last_seed)
    for each seed from first_seed to last_seed
        >set the random seed
        >run the model with the specified parameters
        >append the output of the model in the stored results
    return the results from all replications

function *determine_number_of_replications*(results, error_margin, confidence_interval)
    return the calculated number of replications needed based on all current aggregated results

B.7.2. Process

while there are fewer replications executed than specified by *determine_number_of_replications*
    call *run_script* for the next 8 random seeds
    >extend the current results with the new results
    >increase the number of replications performed by 8
    >recalculate the number of replications needed with the current results
>average out the results across all replications
>store the results in a structured Excel file for each individual crane and rack
>store the results averaged out across cranes and racks in the same Excel file

C Simulation settings

Setting	Value	Explanation
Simulation end time	00:00 2-1-2001	Time until when the simulation runs. Target day is 1-1-2001.
Rack switch-off delay	0 seconds	Time it takes after downtime started to switch off a rack. This means no more new trips are appointed to it.
Rack switch-on delay	1800 seconds	Time it takes after downtime started to turn on a rack again.
Production start mode	1	Mode which determines when production for a trip is started. Explained in section 5.6.
Production start time	06:00	Time at which production starts during target day.
Last produced departure time	09:30, 10:30, 12:00	Departure time of trip on target day until which is produced ahead the day before.
Production offset from departure	36000 seconds	Time offset from departure time when production for a trip can begin.
Retrieval start mode	1	Mode which determines when pallets can move to their outputs. Explained in section 5.6.
Output start	7200 seconds	Time offset before departure time when pallets can move to their outputs.
Loading time factor	0.1	Determines how much longer and shorter it takes to load the first and last pallets respectively.
Loading start	1800 seconds	Time in seconds before departure time when a trip normally starts loading.
Too late threshold	1800 seconds	Time in seconds before departure time after which a pallet is regarded as placed too late at output.
Max pallets in broken buffer	10	Maximum number of pallets of a trip that can be in the broken buffer for a trip to still be rerouted to another one.
Production pool size	30	Number of trips which can be in production concurrently.
Crane storage time	2700 seconds	Time it takes to store a broken crane so that the other crane can continue operation.
Crane retrieval time	1800 seconds	Time it takes to retrieve a repaired crane after it was stored.
Downtime start	5,7,13	Time at which the downtime of the experiment starts.
Downtime duration	3 or 8 hours	Duration of crane downtime in hours.
Policy	0, 1, 2, 3, 4	Policy to test during experiment.

Table C.1: Simulation settings

Setting	Value	Explanation
Horizontal crane acceleration	1.8 m/s^2	-
Vertical crane acceleration	1.8 m/s^2	-
Maximum horizontal crane speed	1.5 m/s	-
Maximum vertical crane speed	1.5 m/s	-
Base depth pickup time	4.8 s	Base time it always takes to pick up a pallet regardless of the depth in which it is placed.
Added depth pickup time	2.85 s	Added time it takes per depth rack location to pick up a pallet.

Table C.2: Crane settings

Setting	Value	Explanation
Input point A1	(2,2,1)	Rack configuration A input point 1 coordinates.
Input point A2	(10,2,1)	Rack configuration A input point 2 coordinates.
Pallet stack input point A	(9,0,-1)	Rack configuration A empty pallet stack input point coordinates.
Pallet stack output point A	(7,2,1)	Rack configuration A empty pallet stack output point coordinates.
Loader division A	0:(0,1), 1:(2,3), 2:(4,5), 3:(6,7), 4:(10,11), 5:(12,13), 6:(14,15)	Rack configuration A outputs to service per loader.
Blocked locations A	(1,2), (1,3), (2,2), (2,3), (6,2), (6,3), (7,2), (7,3), (10,2), (10,3), (11,2), (11,3)	Rack configuration A (x,y) coordinates of locations that are blocked and where pallets cannot be stored.
Failure lane A	(7,0,-1)	Rack configuration A failure lane coordinates.
Input point B1	(3,2,1)	Rack configuration B input point 1 coordinates.
Input point B2	(13,2,1)	Rack configuration B input point 2 coordinates.
Pallet stack input point B	(6,0,-1)	Rack configuration B empty pallet stack input point coordinates.
Pallet stack output point B	(8,2,1)	Rack configuration B empty pallet stack output point coordinates.
Loader division B	0:(0,1), 1:(2,3), 2:(4,5), 3:(8,9), 4:(10,11), 5:(12,13), 6:(14,15)	Rack configuration B outputs to service per loader.
Blocked locations B	(3,2), (3,3), (4,2), (4,3), (8,2), (8,3), (9,2), (9,3), (12,2), (12,3), (13,2), (13,3)	Rack configuration B (x,y) coordinates of locations that are blocked and where pallets cannot be stored.
Failure lane B	(8,0,-1)	Rack configuration B failure lane coordinates.

Table C.3: Rack configuration settings

Setting	Value	Explanation
Number of gravity lanes	14	Number of gravity lanes per rack.
Gravity lane capacity	13 pallets	Maximum amount of pallets that can be stored in a gravity lane.
Number of loaders	7	Number of loaders per rack.
Horizontal location distance	1.87 m	Horizontal distance between two rack locations in meters.
Vertical location distance	2.65 m	Vertical distance between two rack locations in meters.
Pallet stack height	10 pallets	Number of empty pallets needed to form one empty pallet stack.
Rack width	16	Number of pallet locations next to each other in the rack.
Rack height	8	Number of pallet locations above each other in the rack.
Rack depth	3	Number of pallet locations behind each other in the rack.
Number of racks	4	Number of racks in parallel that form the system.
Number of cranes	2	Number of cranes per rack.
Input point capacity	3 pallets	Number of pallets that can wait at an input point before blocking the central conveyor.
Crane dwell points	0:(3,2,0), 1:(12,2,0)	Coordinates where the cranes will wait when passive.
Minimum crane distance	2	Minimum number of rack locations in x coordinates that have to be between the cranes to not collide. A distance of 2 means they could operate on x-location 1 and 4 for example.
Crane service area	0:(0-10), 1:(5-15)	Service area of each crane in x coordinates.

Table C.4: General rack settings

D Quantitative results

D.1. No downtime

	Output delay			Upstream system interference		Added manual work		Robustness							Resilience		
Day	Aggregate trip delay [min]	Max trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Avg crane utilisation [%]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Avg system fill grade [%]	Min margin between departure times [min]	Avg margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
1	701,4	38,2	42,5	0,03	5,19	0,0	0,0	25,1	91,8	54,7	48,0	27,2	151,0	382,0	30,8	37,7	48
2	985,7	40,9	58,1	0,01	3,44	0,0	0,0	33,8	92,1	57,2	54,6	34,1	121,0	296,0	34,3	41,2	96
3	979,9	41,3	56,9	0,01	3,91	0,0	0,0	44,8	92,9	60,1	57,6	39,3	98,0	239,0	36,3	50,6	80

Table D.1: Results without downtime

D.2. Experiment 1: One crane down

D.2.1. Average day in an average week

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1128,5	42,2	0,03	5,30	20,7	20,7	90,2	54,1	48,1	119,0	27,8	37,6	104
1	1121,1	42,4	0,03	4,91	16,8	16,8	89,9	56,7	48,1	151,0	28,3	37,5	88
2	1121,2	41,4	0,03	4,95	15,9	15,9	90,8	51,2	48,1	151,0	28,2	38,0	96
3	1116,5	41,9	0,03	5,52	11,4	11,4	92,0	49,3	48,1	151,0	28,5	38,3	120
4	1123,7	42,7	0,03	4,88	0,0	0,0	95,2	66,1	48,1	151,0	29,9	40,1	104

Table D.2: Results Experiment 1 Day 1 Downtime 05:00 - 09:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2130,1	39,7	0,09	6,61	26,1	26,1	91,5	53,5	48,0	119,0	31,8	41,4	200
1	2165,7	42,9	0,06	5,92	18,9	18,9	88,9	57,2	48,4	151,0	32,7	41,6	200
2	2114,1	39,3	0,05	5,43	20,1	20,1	90,4	53,8	48,5	151,0	32,9	39,8	200
3	2153,5	41,9	0,02	5,07	13,8	13,8	89,3	53,5	48,6	151,0	33,6	42,2	200
4	2158,4	42,7	0,03	5,14	0,0	0,0	95,0	66,5	51,0	151,0	36,0	45,0	200

Table D.3: Results Experiment 1 Day 1 Downtime 05:00 - 14:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1120,4	42,4	0,03	5,02	85,1	50,5	89,0	56,5	48,1	151,0	28,9	33,3	56
1	1120,4	42,4	0,03	5,02	85,1	50,5	89,0	56,5	48,1	151,0	28,9	33,3	56
2	1120,4	42,4	0,03	5,02	85,1	50,5	89,0	56,5	48,1	151,0	28,9	33,3	56
3	1120,4	42,4	0,03	5,02	85,1	50,5	89,0	56,5	48,1	151,0	28,9	33,3	56
4	1133,9	42,5	0,03	5,63	84,8	50,2	90,7	56,3	47,9	151,0	29,3	34,1	48

Table D.4: Results Experiment 1 Day 1 Downtime 07:00 - 11:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2173,4	43,2	0,03	4,89	108,9	58,0	92,4	52,9	48,0	120,0	35,8	39,8	192
1	2184,1	43,6	0,03	5,12	104,7	53,9	90,0	52,4	49,1	151,0	37,4	39,6	200
2	2170,2	43,2	0,03	4,96	103,8	53,0	90,7	53,5	49,6	151,0	38,0	40,1	200
3	2189,7	44,2	0,03	4,99	98,6	50,8	88,7	53,5	50,7	151,0	38,8	42,0	200
4	2188,8	44,5	0,03	5,43	84,5	50,7	92,3	58,3	55,7	151,0	43,2	46,5	176

Table D.5: Results Experiment 1 Day 1 Downtime 07:00 - 16:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	648,3	38,7	0,04	5,15	105,1	63,8	93,2	54,4	48,1	151,0	38,0	40,5	40
1	644,4	38,8	0,04	5,30	102,2	60,9	92,2	55,0	48,9	151,0	38,9	41,1	40
2	642,6	38,6	0,04	5,22	101,7	60,1	89,7	54,6	48,7	151,0	38,8	40,8	64
3	640,7	38,6	0,04	5,21	98,1	56,5	89,3	54,3	50,0	151,0	39,6	39,9	48
4	652,5	38,8	0,04	5,30	88,2	46,8	94,9	55,4	52,5	151,0	42,4	41,5	40

Table D.6: Results Experiment 1 Day 1 Downtime 13:00 - 17:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	653,6	39,5	0,05	5,27	104,4	62,6	91,2	54,2	48,1	151,0	45,7	58,5	48
1	656,8	39,6	0,04	5,05	97,7	56,1	94,1	55,2	51,9	151,0	49,1	61,7	56
2	659,5	39,5	0,04	5,23	97,7	56,3	90,6	54,9	52,4	151,0	49,3	58,1	40
3	653,1	39,5	0,04	5,15	94,6	52,8	92,8	57,8	54,4	151,0	51,6	62,5	48
4	649,0	39,3	0,04	5,44	82,0	41,4	94,7	56,9	63,4	151,0	59,0	65,3	32

Table D.7: Results Experiment 1 Day 1 Downtime 13:00 - 22:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	1309,0	40,9	0,04	5,37	75,0	47,0	91,3	54,2	48,1	135,2	34,7	41,9
1	1315,4	41,6	0,04	5,22	70,9	42,8	90,7	55,5	49,1	151,0	35,9	42,5
2	1304,7	40,7	0,03	5,13	70,7	42,6	90,2	54,1	49,2	151,0	36,0	41,7
3	1312,3	41,4	0,03	5,16	66,9	39,3	90,2	54,2	50,0	151,0	36,8	43,0
4	1317,7	41,7	0,03	5,30	56,6	31,5	93,8	59,9	53,1	151,0	40,0	45,4
Reference minimum	1304,7	40,7	0,03	5,13	0,0	0,0	90,2	54,1	48,1	60	34,7	41,7
Reference maximum	1501,1	100	1	10	84,6	84,6	100	100	65	151,0	60	60

Table D.8: Averaged results Experiment 1 Day 1

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	9,8	10,0	9,9	9,6	2,0	5,0	9,0	10,0	10,0	8,4	10,0	9,9
1	9,5	9,9	10,0	9,8	2,5	5,4	9,6	9,7	9,5	10,0	9,6	9,6
2	10,0	10,0	10,0	10,0	2,5	5,5	10,0	10,0	9,4	10,0	9,5	10,0
3	9,6	9,9	10,0	10,0	2,9	5,8	10,0	10,0	9,0	10,0	9,2	9,3
4	9,4	9,8	10,0	9,7	4,0	6,6	6,7	8,9	7,3	10,0	8,1	8,2

Table D.9: Individual scores Experiment 1 Day 1

D.2.2. Peak day in an average week

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1091,6	58,3	0,04	5,22	24,9	24,9	99,0	94,1	54,6	61,0	32,3	42,3	144
1	1041,6	57,0	0,02	4,29	20,0	20,0	96,9	79,7	54,6	121,0	32,9	43,3	144
2	1040,5	56,4	0,02	4,61	20,1	20,1	96,0	78,1	54,6	121,0	32,7	43,1	136
3	1036,9	56,7	0,02	4,05	15,6	15,6	93,8	69,0	54,6	121,0	33,4	44,5	136
4	1048,1	57,3	0,01	3,45	2,1	2,1	93,6	62,5	54,6	117,0	34,6	46,9	160

Table D.10: Results Experiment 1 Day 2 Downtime 05:00 - 09:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1751,9	58,4	0,09	6,21	23,8	23,8	100,0	123,8	54,6	58,0	35,0	46,6	200
1	1721,4	58,4	0,02	3,90	19,3	19,3	97,9	81,6	54,6	63,0	35,9	45,6	200
2	1673,7	56,0	0,06	5,27	19,2	19,2	96,5	77,4	54,6	120,0	36,8	45,2	200
3	1721,5	58,5	0,03	3,97	16,0	16,0	92,5	68,7	54,6	118,0	37,7	46,2	200
4	1725,4	59,3	0,02	3,68	0,0	0,0	96,7	72,5	54,9	90,0	40,7	50,6	200

Table D.11: Results Experiment 1 Day 2 Downtime 05:00 - 14:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1208,6	59,6	0,05	5,50	136,9	68,7	92,3	57,1	54,6	121,0	33,3	41,8	88
1	1208,6	59,6	0,05	5,50	136,9	68,7	92,3	57,1	54,6	121,0	33,3	41,8	88
2	1184,0	59,0	0,03	4,41	130,1	68,3	91,6	57,0	54,6	121,0	33,4	42,1	64
3	1177,3	58,1	0,02	4,36	126,3	68,5	92,1	57,1	54,6	121,0	33,4	42,1	80
4	1187,2	59,4	0,02	3,86	108,2	68,6	91,8	56,9	54,6	121,0	34,1	42,3	72

Table D.12: Results Experiment 1 Day 2 Downtime 07:00 - 11:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2253,8	60,1	0,03	4,33	133,0	69,0	91,9	61,0	54,6	121,0	37,0	44,2	200
1	2243,9	59,4	0,03	3,98	128,4	69,0	92,1	57,5	54,6	121,0	38,6	45,2	200
2	2249,1	60,3	0,02	3,98	126,2	69,0	92,2	57,6	54,6	121,0	39,0	45,5	200
3	2277,6	61,4	0,02	3,78	123,8	69,0	92,2	57,6	54,6	121,0	40,4	46,5	200
4	2255,2	61,5	0,02	3,91	113,1	69,0	93,5	64,9	56,3	121,0	43,8	47,7	200

Table D.13: Results Experiment 1 Day 2 Downtime 07:00 - 16:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1071,0	54,1	0,02	3,93	116,1	61,9	93,0	58,0	54,6	121,0	38,9	41,1	72
1	1067,5	54,2	0,02	3,75	113,9	59,3	92,7	58,4	54,6	121,0	39,5	41,5	56
2	1073,3	54,1	0,02	3,86	112,9	58,5	92,8	57,9	54,6	121,0	39,6	41,2	64
3	1073,4	54,1	0,02	3,77	110,8	56,5	92,5	58,9	54,6	121,0	40,4	41,6	48
4	1070,4	54,2	0,02	3,59	98,9	54,7	92,8	58,3	55,5	121,0	42,5	43,3	56

Table D.14: Results Experiment 1 Day 2 Downtime 13:00 - 17:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1492,9	56,0	0,02	3,92	115,8	60,9	93,0	57,7	54,6	121,0	48,2	61,5	72
1	1486,2	56,0	0,02	3,89	112,5	57,4	92,8	57,9	54,6	121,0	50,6	61,6	64
2	1492,7	56,1	0,02	3,74	109,7	54,9	93,0	57,7	54,8	121,0	51,5	62,5	72
3	1486,9	56,0	0,02	3,64	106,6	55,3	92,7	58,3	56,1	121,0	53,4	65,3	56
4	1481,9	56,0	0,02	3,73	94,2	55,1	92,8	57,9	64,0	121,0	60,4	70,7	64

Table D.15: Results Experiment 1 Day 2 Downtime 13:00 - 22:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	1478,3	57,7	0,04	4,85	91,7	51,5	94,9	75,3	54,6	100,5	37,5	46,2
1	1461,5	57,5	0,03	4,22	88,5	48,9	94,1	65,4	54,6	111,3	38,5	46,5
2	1452,2	57,0	0,03	4,31	86,4	48,3	93,7	64,3	54,6	120,8	38,8	46,6
3	1462,3	57,5	0,02	3,93	83,2	46,8	92,6	61,6	54,8	120,5	39,8	47,7
4	1461,4	57,9	0,02	3,70	69,4	41,6	93,5	62,2	56,6	115,2	42,7	50,2
Reference minimum	1452,2	57,0	0,01	3,61	0,0	0,0	92,6	61,6	54,6	60	37,5	46,2
Reference maximum	2221,0	70	1	10	102,6	102,6	100	120	70	120,8	60	60

Table D.16: Averaged results Experiment 1 Day 2

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	9,7	9,5	9,7	8,3	2,0	5,5	7,3	7,9	10,0	7,0	10,0	10,0
1	9,9	9,7	9,9	9,1	2,2	5,7	8,2	9,4	10,0	8,6	9,6	9,8
2	10,0	10,0	9,9	9,0	2,4	5,8	8,7	9,6	10,0	10,0	9,4	9,8
3	9,9	9,7	9,9	9,6	2,7	5,9	10,0	10,0	9,8	10,0	9,1	9,0
4	9,9	9,3	10,0	9,9	3,9	6,4	8,9	9,9	8,8	9,2	7,9	7,4

Table D.17: Individual scores Experiment 1 Day 2

D.2.3. Peak day in a peak week

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1611,6	61,3	0,04	5,06	26,4	26,4	100,0	161,5	57,7	40,0	35,7	50,9	200
1	1552,4	61,4	0,03	4,27	21,5	21,5	99,8	129,4	57,7	56,0	36,0	50,4	200
2	1561,3	61,5	0,03	4,45	20,4	20,4	99,6	122,6	57,7	52,0	36,0	50,4	200
3	1479,5	59,4	0,02	4,04	16,8	16,8	97,7	99,3	57,7	70,0	36,4	51,5	200
4	1413,4	58,8	0,02	3,93	4,2	4,2	96,3	88,0	57,6	82,0	38,0	53,7	152

Table D.18: Results Experiment 1 Day 3 Downtime 05:00 - 09:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2692,0	62,1	0,18	7,66	24,6	24,6	100,0	252,6	57,7	32,0	36,3	64,7	200
1	2400,9	61,1	0,13	7,33	19,5	19,5	100,0	168,9	57,7	61,0	37,9	53,8	200
2	2309,0	57,9	0,09	6,14	17,6	17,6	100,0	160,6	57,7	76,0	38,4	54,9	200
3	2317,0	58,6	0,09	6,20	13,9	13,9	99,8	136,0	57,7	87,0	39,1	55,9	200
4	2373,7	62,2	0,03	4,04	0,3	0,3	99,9	189,6	57,7	60,0	43,2	65,4	200

Table D.19: Results Experiment 1 Day 3 Downtime 05:00 - 14:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1313,0	57,7	0,04	4,40	114,0	64,6	99,7	95,8	57,7	60,0	35,9	51,8	88
1	1288,0	56,8	0,03	4,30	112,2	62,8	99,7	95,7	57,6	61,0	36,1	52,0	80
2	1293,2	57,5	0,03	4,29	111,3	61,8	99,7	96,6	57,7	75,0	36,2	52,0	96
3	1241,7	56,5	0,03	4,21	107,4	57,8	99,8	96,6	57,7	89,0	36,6	53,2	96
4	1134,1	54,6	0,03	4,17	94,3	49,5	99,8	96,3	57,7	74,0	37,4	54,7	88

Table D.20: Results Experiment 1 Day 3 Downtime 07:00 - 11:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2377,1	57,8	0,07	5,30	117,6	64,3	99,6	128,3	57,7	60,0	36,6	55,6	200
1	2332,1	55,7	0,05	4,73	114,4	61,2	99,7	96,8	57,7	61,0	38,1	49,3	200
2	2336,4	56,5	0,05	4,60	113,4	60,2	99,6	96,9	57,7	75,0	38,7	49,8	200
3	2352,6	57,5	0,04	4,42	110,8	57,6	99,6	97,1	57,7	89,0	39,5	52,8	200
4	2345,9	58,5	0,04	4,19	98,7	53,2	99,9	112,8	57,7	74,0	43,5	59,0	200

Table D.21: Results Experiment 1 Day 3 Downtime 07:00 - 16:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1443,2	55,2	0,03	4,22	117,5	62,6	99,7	113,8	57,7	98,0	38,0	46,9	64
1	1452,3	55,3	0,03	4,13	114,5	60,2	99,6	112,7	57,6	98,0	38,5	47,4	40
2	1443,5	55,1	0,03	4,04	114,0	59,5	99,7	110,7	57,6	98,0	38,7	47,2	48
3	1439,6	55,3	0,03	4,14	110,2	55,3	99,7	110,3	57,7	98,0	39,2	47,4	72
4	1439,3	55,2	0,02	4,07	102,9	54,9	99,6	104,3	57,7	98,0	41,0	47,8	72

Table D.22: Results Experiment 1 Day 3 Downtime 13:00 - 17:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2592,3	58,6	0,04	4,42	115,9	60,3	99,6	115,1	57,7	98,0	44,6	57,6	112
1	2586,3	58,5	0,03	4,04	113,6	58,0	99,6	114,4	57,7	98,0	46,6	53,4	112
2	2588,0	58,4	0,03	4,11	110,5	55,6	99,6	113,4	57,7	98,0	47,2	54,1	104
3	2586,8	58,5	0,03	4,24	108,0	55,6	99,6	110,4	57,7	98,0	49,2	55,4	104
4	2593,2	58,7	0,02	3,99	95,7	55,6	99,5	103,6	59,2	98,0	55,5	60,8	112

Table D.23: Results Experiment 1 Day 3 Downtime 13:00 - 22:15

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	2004,9	58,8	0,07	5,17	86,0	50,5	99,8	144,5	57,7	64,7	37,9	54,6
1	1935,3	58,1	0,05	4,80	82,6	47,2	99,7	119,7	57,7	72,5	38,9	51,1
2	1921,9	57,8	0,04	4,60	81,2	45,8	99,7	116,8	57,7	79,0	39,2	51,4
3	1902,9	57,6	0,04	4,54	77,9	42,8	99,4	108,3	57,7	88,5	40,0	52,7
4	1883,3	58,0	0,03	4,07	66,0	36,3	99,2	115,8	57,9	81,0	43,1	56,9
Reference minimum	1883,3	57,6	0,02	4,01	0,0	0,0	90	103,9	57,7	60	37,9	51,1
Reference maximum	2853,5	100	1	10	89,8	89,8	100	180	70	88,5	55	70

Table D.24: Averaged results Experiment 1 Day 3

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	8,9	9,8	9,6	8,3	1,4	4,9	1,2	5,2	10,0	2,5	10,0	8,3
1	9,5	9,9	9,8	8,8	1,7	5,3	1,2	8,1	10,0	4,9	9,5	10,0
2	9,6	10,0	9,8	9,1	1,9	5,4	1,3	8,5	10,0	7,0	9,3	9,8
3	9,8	10,0	9,9	9,2	2,2	5,7	1,6	9,5	10,0	10,0	8,9	9,2
4	10,0	9,9	10,0	9,9	3,4	6,4	1,7	8,6	9,8	7,6	7,2	7,2

Table D.25: Individual scores Experiment 1 Day 3

D.3. Experiment 2: Both cranes down

D.3.1. Average day in an average week

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	935,5	43,5	0,03	5,88	0,0	0,0	89,7	54,3	47,9	151,0	29,8	40,2	64
1	935,5	43,5	0,03	5,88	0,0	0,0	89,7	54,3	47,9	151,0	29,8	40,2	64
2	935,5	43,5	0,03	5,88	0,0	0,0	89,7	54,3	47,9	151,0	29,8	40,2	64
3	935,5	43,5	0,03	5,88	0,0	0,0	89,7	54,3	47,9	151,0	29,8	40,2	64
4	935,5	43,5	0,03	5,88	0,0	0,0	89,7	54,3	47,9	151,0	29,8	40,2	64

Table D.26: Results Experiment 2 Day 1 Downtime 05:00 - 08:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2152,3	43,2	0,04	5,13	0,0	0,0	96,3	69,2	50,3	151,0	36,0	45,5	200
1	2152,3	43,2	0,04	5,13	0,0	0,0	96,3	69,2	50,3	151,0	36,0	45,5	200
2	2152,3	43,2	0,04	5,13	0,0	0,0	96,3	69,2	50,3	151,0	36,0	45,5	200
3	2152,3	43,2	0,04	5,13	0,0	0,0	96,3	69,2	50,3	151,0	36,0	45,5	200
4	2152,3	43,2	0,04	5,13	0,0	0,0	96,3	69,2	50,3	151,0	36,0	45,5	200

Table D.27: Results Experiment 2 Day 1 Downtime 05:00 - 13:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1071,5	42,7	11,09	135,15	0,0	0,0	90,8	104,7	48,1	151,0	29,0	37,5	168
1	931,6	40,3	0,09	6,63	137,2	137,2	99,4	115,7	48,0	151,0	28,9	35,0	112
2	960,9	41,1	0,04	6,13	80,3	58,0	97,1	91,3	48,0	105,0	31,3	37,2	200
3	929,6	40,5	0,03	5,28	137,2	52,2	93,3	67,7	48,2	151,0	28,9	34,2	112
4	944,1	40,6	0,03	5,94	80,3	58,0	94,5	69,3	48,0	150,0	29,7	34,3	200

Table D.28: Results Experiment 2 Day 1 Downtime 07:00 - 10:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	3281,5	47,4	39,19	136,48	0,0	0,0	92,3	101,4	55,6	151,0	43,2	45,9	200
1	2844,3	47,3	0,10	6,89	137,3	137,3	99,5	117,9	55,8	151,0	43,2	45,8	200
2	2868,6	47,5	0,05	6,20	80,3	58,0	97,2	92,5	55,8	105,0	43,2	43,9	200
3	2835,1	47,3	0,04	5,23	137,3	52,1	94,2	71,9	55,6	151,0	43,2	46,0	200
4	2856,7	47,3	0,04	5,90	80,4	58,0	94,9	71,5	55,7	150,0	43,2	45,3	200

Table D.29: Results Experiment 2 Day 1 Downtime 07:00 - 15:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	703,9	42,4	9,53	109,65	0,0	0,0	91,4	70,6	52,5	151,0	42,2	40,6	104
1	650,4	39,4	0,03	5,29	115,3	115,3	93,2	54,9	52,3	151,0	42,1	40,7	80
2	668,3	39,9	0,03	5,22	47,2	39,7	93,8	69,8	58,2	133,0	48,1	43,9	200
3	658,5	39,5	0,03	5,34	115,2	46,7	92,1	54,6	52,3	151,0	42,1	40,6	80
4	656,3	39,7	0,03	5,17	47,1	39,8	90,9	59,3	54,1	151,0	44,4	42,3	200

Table D.30: Results Experiment 2 Day 1 Downtime 13:00 - 16:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	861,7	44,2	32,81	115,00	0,0	0,0	96,2	68,4	64,0	151,0	59,0	64,1	200
1	695,9	41,3	0,04	5,25	117,5	117,5	95,2	57,4	64,1	151,0	59,1	63,7	200
2	705,7	41,7	0,03	5,17	47,6	40,1	96,4	69,5	64,5	133,0	61,8	69,3	200
3	693,2	41,3	0,03	5,24	117,5	47,2	95,7	56,9	64,0	151,0	59,1	64,0	200
4	700,4	41,6	0,03	5,21	47,5	40,1	95,2	59,0	64,0	151,0	60,3	68,2	200

Table D.31: Results Experiment 2 Day 1 Downtime 13:00 - 21:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	1501,1	43,9	15,45	84,55	0,0	0,0	92,8	78,1	53,1	151,0	39,9	45,6
1	1368,3	42,5	0,05	5,84	84,6	84,6	95,5	78,2	53,1	151,0	39,9	45,2
2	1381,9	42,8	0,04	5,62	42,6	32,7	95,1	74,4	54,1	129,7	41,7	46,7
3	1367,4	42,5	0,03	5,35	84,5	33,0	93,5	62,4	53,1	151,0	39,9	45,1
4	1374,2	42,6	0,03	5,54	42,6	32,7	93,6	63,7	53,4	150,7	40,6	46,0
Reference minimum	1304,7	40,7	0,03	5,13	0,0	0,0	90,2	54,1	48,1	60	34,7	41,7
Reference maximum	1501,1	100	1	10	84,6	84,6	100	100	65	151,0	60	60

Table D.32: Averaged results Experiment 2 Day 1

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	1,0	9,5	1,0	1,0	10,0	10,0	7,6	5,3	7,3	10,0	8,2	8,1
1	7,1	9,7	9,8	8,7	1,0	1,0	5,1	5,3	7,3	10,0	8,2	8,3
2	6,5	9,7	10,0	9,1	5,5	6,5	5,5	6,0	6,8	7,9	7,5	7,5
3	7,1	9,7	10,0	9,6	1,0	6,5	6,9	8,4	7,3	10,0	8,2	8,3
4	6,8	9,7	10,0	9,3	5,5	6,5	6,9	8,1	7,2	10,0	7,9	7,9

Table D.33: Individual scores Experiment 2 Day 1

D.3.2. Peak day in an average week

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1336,9	61,0	0,01	3,42	0,0	0,0	93,9	62,3	54,5	91,0	34,8	45,6	112
1	1336,9	61,0	0,01	3,42	0,0	0,0	93,9	62,3	54,5	91,0	34,8	45,6	112
2	1336,9	61,0	0,01	3,42	0,0	0,0	93,9	62,3	54,5	91,0	34,8	45,6	112
3	1336,9	61,0	0,01	3,42	0,0	0,0	93,9	62,3	54,5	91,0	34,8	45,6	112
4	1336,9	61,0	0,01	3,42	0,0	0,0	93,9	62,3	54,5	91,0	34,8	45,6	112

Table D.34: Results Experiment 2 Day 2 Downtime 05:00 - 08:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	3066,9	62,3	0,02	3,91	0,0	0,0	96,8	74,2	54,9	90,0	40,6	50,8	200
1	3066,9	62,3	0,02	3,91	0,0	0,0	96,8	74,2	54,9	90,0	40,6	50,8	200
2	3066,9	62,3	0,02	3,91	0,0	0,0	96,8	74,2	54,9	90,0	40,6	50,8	200
3	3066,9	62,3	0,02	3,91	0,0	0,0	96,8	74,2	54,9	90,0	40,6	50,8	200
4	3066,9	62,3	0,02	3,91	0,0	0,0	96,8	74,2	54,9	90,0	40,6	50,8	200

Table D.35: Results Experiment 2 Day 2 Downtime 05:00 - 13:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1214,2	58,0	11,38	139,14	0,0	0,0	91,9	98,7	54,6	118,0	34,1	43,7	200
1	1090,2	54,5	0,03	4,20	143,5	143,5	93,5	60,3	54,6	118,0	34,1	43,6	88
2	1110,2	55,2	0,02	3,93	68,1	36,0	95,8	74,5	54,6	33,0	38,4	50,5	200
3	1085,6	54,6	0,01	3,66	143,6	55,0	92,5	57,8	54,6	118,0	34,1	43,9	88
4	1087,3	54,7	0,02	3,69	68,1	36,0	92,7	59,0	54,6	87,0	35,7	44,3	200

Table D.36: Results Experiment 2 Day 2 Downtime 07:00 - 10:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	3367,6	62,3	43,27	150,82	0,0	0,0	94,2	99,1	57,5	118,0	44,8	48,9	200
1	2900,2	62,2	0,04	4,26	153,2	153,2	96,5	75,2	57,5	118,0	44,7	48,6	88
2	2922,1	62,6	0,02	3,81	67,5	35,0	99,0	94,9	57,5	33,0	44,9	52,2	120
3	2911,7	62,2	0,02	3,83	153,3	58,6	94,6	68,6	57,6	118,0	44,7	48,4	80
4	2902,2	62,0	0,02	3,66	68,3	36,1	94,5	64,6	57,5	87,0	44,8	49,5	200

Table D.37: Results Experiment 2 Day 2 Downtime 07:00 - 15:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1537,1	58,0	11,63	139,36	0,0	0,0	91,8	130,3	55,7	121,0	42,5	44,2	200
1	1321,6	54,6	0,06	4,71	144,2	144,2	98,7	103,8	56,0	121,0	42,5	46,7	96
2	1334,8	55,4	0,02	3,65	26,3	15,1	97,5	85,5	70,0	81,0	58,2	53,0	200
3	1308,6	54,5	0,01	3,47	144,1	60,5	92,2	56,9	55,7	121,0	42,5	44,8	104
4	1327,8	55,4	0,01	3,48	26,2	15,0	92,3	57,7	59,7	121,0	48,5	46,4	200

Table D.38: Results Experiment 2 Day 2 Downtime 13:00 - 16:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	2803,4	59,2	47,67	169,99	0,0	0,0	91,9	116,3	64,3	121,0	59,9	71,6	200
1	2603,2	56,2	0,06	4,70	174,6	174,6	99,5	121,9	64,1	121,0	59,7	71,5	200
2	2620,6	56,9	0,02	3,72	26,5	15,3	98,3	100,7	74,0	81,0	70,6	82,5	200
3	2599,3	56,3	0,02	3,56	174,6	75,0	93,9	64,4	64,2	121,0	59,8	71,9	200
4	2614,2	56,7	0,01	3,52	26,5	15,3	92,4	59,0	66,4	121,0	63,3	72,2	200

Table D.39: Results Experiment 2 Day 2 Downtime 13:00 - 21:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	2221,0	60,1	19,00	101,10	0,0	0,0	93,4	96,8	56,9	109,8	42,8	50,8
1	2053,2	58,5	0,04	4,20	102,6	102,6	96,5	83,0	56,9	109,8	42,7	51,1
2	2065,2	58,9	0,02	3,74	31,4	16,9	96,9	82,0	60,9	68,2	47,9	55,7
3	2051,5	58,5	0,02	3,64	102,6	41,5	94,0	64,0	56,9	109,8	42,7	50,9
4	2055,9	58,7	0,01	3,61	31,5	17,1	93,8	62,8	58,0	99,5	44,6	51,5
Reference minimum	1452,2	57,0	0,01	3,61	0,0	0,0	92,6	61,6	54,6	60	37,5	46,2
Reference maximum	2221,0	70	1	10	102,6	102,6	100	120	70	120,8	60	60

Table D.40: Averaged results Experiment 2 Day 2

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience	
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]
0	1,0	7,8	1,0	1,0	10,0	10,0	9,0	4,6	8,6	8,4	7,9	7,0
1	3,0	9,0	9,8	9,2	1,0	1,0	5,3	6,7	8,6	8,4	7,9	6,8
2	2,8	8,7	10,0	9,8	7,2	8,5	4,8	6,9	6,3	2,2	5,8	3,8
3	3,0	9,0	10,0	10,0	1,0	6,4	8,4	9,6	8,6	8,4	7,9	7,0
4	2,9	8,8	10,0	10,0	7,2	8,5	8,6	9,8	8,0	6,8	7,1	6,6

Table D.41: Individual scores Experiment 2 Day 2

D.3.3. Peak day in a peak week

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1601,9	60,9	0,02	3,98	0,0	0,0	97,2	103,1	57,7	73,0	38,0	54,1	192
1	1601,9	60,9	0,02	3,98	0,0	0,0	97,2	103,1	57,7	73,0	38,0	54,1	192
2	1601,9	60,9	0,02	3,98	0,0	0,0	97,2	103,1	57,7	73,0	38,0	54,1	192
3	1601,9	60,9	0,02	3,98	0,0	0,0	97,2	103,1	57,7	73,0	38,0	54,1	192
4	1601,9	60,9	0,02	3,98	0,0	0,0	97,2	103,1	57,7	73,0	38,0	54,1	192

Table D.42: Results Experiment 2 Day 3 Downtime 05:00 - 08:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	4740,2	64,0	0,04	4,21	0,0	0,0	100,0	179,4	57,7	60,0	43,2	64,5	200
1	4740,2	64,0	0,04	4,21	0,0	0,0	100,0	179,4	57,7	60,0	43,2	64,5	200
2	4740,2	64,0	0,04	4,21	0,0	0,0	100,0	179,4	57,7	60,0	43,2	64,5	200
3	4740,2	64,0	0,04	4,21	0,0	0,0	100,0	179,4	57,7	60,0	43,2	64,5	200
4	4740,2	64,0	0,04	4,21	0,0	0,0	100,0	179,4	57,7	60,0	43,2	64,5	200

Table D.43: Results Experiment 2 Day 3 Downtime 05:00 - 13:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1460,1	57,6	9,53	116,21	0,0	0,0	98,4	137,6	58,1	74,0	38,1	54,8	176
1	1329,3	57,7	0,04	4,51	121,0	121,0	99,7	129,7	58,2	74,0	38,1	54,8	96
2	1353,4	58,8	0,03	4,02	46,9	36,5	98,7	113,0	58,7	40,0	45,6	56,4	200
3	1331,8	57,4	0,02	3,92	119,5	49,8	99,1	81,1	57,7	74,0	38,1	55,5	104
4	1313,3	57,8	0,02	4,00	46,9	36,5	98,8	95,5	58,2	56,0	40,8	55,9	200

Table D.44: Results Experiment 2 Day 3 Downtime 07:00 - 10:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	4417,7	65,6	39,43	130,90	0,0	0,0	99,7	121,6	58,2	74,0	44,4	59,9	200
1	4037,5	65,6	0,05	4,55	133,7	133,7	100,0	132,5	58,2	74,0	44,4	59,8	200
2	4080,4	67,7	0,04	4,14	46,9	36,5	100,0	206,3	58,2	40,0	44,5	62,3	200
3	4043,9	65,6	0,03	3,97	133,4	57,2	99,7	102,4	57,7	74,0	44,4	59,9	200
4	4052,4	66,5	0,03	4,05	46,9	36,5	99,9	140,1	58,2	56,0	44,4	60,9	200

Table D.45: Results Experiment 2 Day 3 Downtime 07:00 - 15:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	1549,7	61,2	10,16	125,72	0,0	0,0	93,6	164,5	58,2	98,0	42,0	47,5	200
1	1340,4	60,2	0,08	5,35	130,0	130,0	100,0	211,4	58,2	98,0	42,1	50,0	112
2	1340,6	60,5	0,03	4,07	15,9	8,4	99,4	107,0	65,9	76,0	55,6	63,1	200
3	1326,6	59,0	0,02	4,03	131,7	55,6	97,0	77,6	57,7	98,0	42,1	47,5	104
4	1328,6	60,2	0,02	3,94	15,9	8,4	95,3	66,4	58,2	97,0	46,6	52,0	200

Table D.46: Results Experiment 2 Day 3 Downtime 13:00 - 16:00

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Policy	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
0	3351,5	66,3	44,70	150,05	0,0	0,0	95,3	135,9	60,3	98,0	56,2	67,8	200
1	3053,7	62,9	0,09	5,36	154,0	154,0	100,0	233,4	60,2	98,0	56,2	67,4	80
2	3069,2	63,0	0,03	4,12	15,9	8,4	99,7	126,5	69,0	76,0	63,5	92,4	200
3	3048,9	61,7	0,02	3,97	153,9	65,7	97,7	79,9	60,2	98,0	56,3	68,2	64
4	3057,7	62,5	0,02	4,04	15,4	8,4	97,0	78,2	61,6	97,0	58,9	72,7	200

Table D.47: Results Experiment 2 Day 3 Downtime 13:00 - 21:00

Policy	Output delay: Aggregate trip delay [min]	Output delay: #Trips with delay [trips]	Upstream system interference: Avg #pallets on conveyor [pallets]	Upstream system interference: Max #pallets on conveyor [pallets]	Added manual work: Total #direct unloads [pallets]	Added manual work: Max #direct unloads one rack [pallets]	Robustness: Max crane utilisation [%/h]	Robustness: Longest continuous utilisation [min]	Robustness: Max system fill grade [%]	Robustness: Min margin between departure times [min]	Resilience: Max avg fill grade since downtime stop [%pt]	Resilience: Max avg crane utilisation since downtime stop [%]
0	2853,5	62,6	17,31	88,51	0,0	0,0	97,4	140,4	58,4	79,5	43,7	58,1
1	2683,8	61,9	0,05	4,66	89,8	89,8	99,5	164,9	58,4	79,5	43,7	58,4
2	2697,6	62,5	0,03	4,09	20,9	15,0	99,2	139,2	61,2	60,8	48,4	65,5
3	2682,2	61,4	0,02	4,01	89,7	38,0	98,4	103,9	58,1	79,5	43,7	58,3
4	2682,4	62,0	0,02	4,03	20,8	15,0	98,0	110,5	58,6	73,2	45,3	60,0
Reference minimum	1883,3	57,6	0,02	4,01	0,0	0,0	90	103,9	57,7	60	37,9	51,1
Reference maximum	2853,5	100	1	10	89,8	89,8	100	180	70	88,5	55	70

Table D.48: Averaged results Experiment 2 Day 3

Policy	Output delay: Aggregate trip delay [min]	Output delay: #Trips with delay [trips]	Upstream system interference: Avg #pallets on conveyor [pallets]	Upstream system interference: Max #pallets on conveyor [pallets]	Added manual work: Total #direct unloads [pallets]	Added manual work: Max #direct unloads one rack [pallets]	Robustness: Max crane utilisation [%/h]	Robustness: Longest continuous utilisation [min]	Robustness: Max system fill grade [%]	Robustness: Min margin between departure times [min]	Resilience: Max avg fill grade since downtime stop [%pt]	Resilience: Max avg crane utilisation since downtime stop [%]
0	1,0	8,9	1,0	1,0	10,0	10,0	3,4	5,7	9,5	7,2	7,0	6,6
1	2,6	9,1	9,7	9,0	1,0	1,0	1,5	2,8	9,5	7,2	7,0	6,5
2	2,4	9,0	10,0	9,9	7,9	8,5	1,7	5,8	7,4	1,3	4,5	3,2
3	2,6	9,2	10,0	10,0	1,0	6,2	2,4	10,0	9,7	7,2	7,0	6,6
4	2,6	9,1	10,0	10,0	7,9	8,5	2,8	9,2	9,3	5,2	6,1	5,7

Table D.49: Individual scores Experiment 2 Day 3

D.4. 1-hour downtime

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Downtime scenario	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
Day 1: 05:00 - 07:15	815,2	41,6	0,03	4,94	22,9	22,9	88,5	52,7	48,2	119,0	27,0	37,3	48
Day 1: 07:00 - 09:15	796,9	41,5	0,03	5,07	73,1	56,1	88,2	56,8	48,2	151,0	27,9	37,9	56
Day 1: 13:00 - 15:15	649,3	38,7	0,04	5,28	79,4	63,4	88,5	55,7	48,2	151,0	33,6	41,8	40
Day 2: 05:00 - 07:15	1151,8	60,4	0,02	4,10	25,1	25,1	92,4	65,5	54,6	121,0	31,3	43,5	96
Day 2: 07:00 - 09:15	974,3	56,9	0,02	4,14	85,2	64,4	93,2	61,6	54,6	121,0	32,3	42,1	64
Day 2: 13:00 - 15:15	997,6	54,3	0,02	3,70	87,8	60,7	93,4	57,8	54,6	121,0	36,0	44,4	88
Day 3: 05:00 - 07:15	1181,5	60,5	0,04	5,07	25,7	25,7	97,7	92,4	57,7	90,0	35,2	52,7	104
Day 3: 07:00 - 09:15	1046,8	55,0	0,03	4,27	92,8	66,1	99,6	96,5	57,7	68,0	35,4	50,9	200
Day 3: 13:00 - 15:15	1153,4	55,3	0,03	4,42	87,8	59,8	92,9	78,3	57,7	98,0	36,4	51,5	72
Average Day 1	753,8	40,6	0,03	5,09	58,5	47,4	88,4	55,0	48,2	140,3	29,5	39,0	
Average Day 2	1041,2	57,2	0,02	3,98	66,0	50,1	93,0	61,6	54,6	121,0	33,2	43,3	
Average Day 3	1127,2	56,9	0,03	4,58	68,8	50,5	96,7	89,1	57,7	85,3	35,7	51,7	

Table D.50: Results with 1 hour of repair time Experiment 1

	Output delay		Upstream system interference		Added manual work		Robustness				Resilience		
Downtime scenario	Aggregate trip delay [min]	#Trips with delay [trips]	Avg #pallets on conveyor [pallets]	Max #pallets on conveyor [pallets]	Total #direct unloads [pallets]	Max #direct unloads one rack [pallets]	Max crane utilisation [%/h]	Longest continuous utilisation [min]	Max system fill grade [%]	Min margin between departure times [min]	Max avg fill grade since downtime stop [%pt]	Max avg crane utilisation since downtime stop [%]	Replications
Day 1: 05:00 - 06:00	720,5	42,1	0,03	6,00	0,0	0,0	89,9	55,1	48,2	151,0	28,7	39,8	74
Day 1: 07:00 - 08:00	700,0	39,3	0,03	5,69	48,9	27,1	92,7	55,5	48,2	150,0	28,8	38,5	200
Day 1: 13:00 - 14:00	666,3	40,4	0,03	5,39	28,0	20,5	91,0	59,9	49,9	151,0	36,0	41,4	88
Day 2: 05:00 - 06:00	1000,1	58,3	0,01	3,59	0,0	0,0	93,4	60,7	54,6	118,0	33,9	47,0	80
Day 2: 07:00 - 08:00	923,4	54,3	0,01	3,62	50,6	31,8	92,4	58,1	54,6	97,0	34,0	45,5	200
Day 2: 13:00 - 14:00	1005,0	55,9	0,01	3,46	17,4	11,1	92,0	57,0	55,2	121,0	41,2	46,6	200
Day 3: 05:00 - 06:00	1049,3	57,8	0,02	3,77	0,0	0,0	95,9	90,1	57,6	87,0	37,4	55,8	56
Day 3: 07:00 - 08:00	1016,7	56,8	0,02	3,87	42,8	35,7	95,0	68,7	57,7	72,0	38,2	52,1	184
Day 3: 13:00 - 14:00	971,8	55,8	0,02	3,89	15,0	8,4	94,9	72,5	57,7	98,0	40,9	56,3	64
Average Day 1	695,6	40,6	0,03	5,69	25,6	15,8	91,2	56,9	48,8	150,7	31,2	39,9	
Average Day 2	976,2	56,2	0,01	3,55	22,7	14,3	92,6	58,6	54,8	112,0	36,4	46,4	
Average Day 3	1012,6	56,8	0,02	3,84	19,3	14,7	95,3	77,1	57,7	85,7	38,8	54,8	

Table D.51: Results with 1 hour of repair time Experiment 2

D.5. Sensitivity analysis KPI weights

	Split ranking [%]					Combined ranking [%]										Avg
Policy	1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	**score**
One crane down - 100% capacity	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	68,5	31,5	0,0	0,0	0,0	0,0	7,93
One crane down - 75% capacity	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	0,0	0,0	8,28
One crane down - 67% capacity	0,0	24,7	75,3	0,0	0,0	0,0	24,7	75,3	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,42
One crane down - 50% capacity	95,7	4,3	0,0	0,0	0,0	95,7	4,3	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,55
One crane down - 0% capacity	4,9	71,0	24,1	0,0	0,0	4,9	71,0	24,1	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,47
Both cranes down - No action	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	100,0	5,36
Both cranes down - Unload at partner	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	100,0	0,0	6,29
Both cranes down - Store in partner	0,0	72,8	27,2	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	72,8	27,2	0,0	0,0	7,33
Both cranes down - Unload spread out	0,0	28,4	71,6	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	28,4	71,6	0,0	0,0	7,23
Both cranes down - Store spread out	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	31,5	68,5	0,0	0,0	0,0	0,0	7,86

Table D.52: Ranking of policies for all weight combinations with an allowed weight deviation of 1

	Split ranking [%]					Combined ranking [%]										Avg
Policy	1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	**score**
One crane down - 100% capacity	0,0	0,0	0,0	0,0	100,0	0,0	0,0	0,0	0,0	60,5	38,6	0,9	0,0	0,0	0,0	7,94
One crane down - 75% capacity	0,0	0,0	7,5	92,5	0,0	0,0	0,0	7,5	90,1	2,4	0,0	0,0	0,0	0,0	0,0	8,29
One crane down - 67% capacity	0,0	41,8	58,2	0,0	0,0	0,0	41,8	58,0	0,2	0,0	0,0	0,0	0,0	0,0	0,0	8,41
One crane down - 50% capacity	89,5	10,5	0,0	0,0	0,0	89,5	10,5	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	8,54
One crane down - 0% capacity	10,7	48,1	33,9	7,3	0,0	10,7	48,1	33,9	7,3	0,0	0,0	0,0	0,0	0,0	0,0	8,44
Both cranes down - No action	0,0	0,0	0,0	7,7	92,3	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	7,7	92,3	5,40
Both cranes down - Unload at partner	0,0	0,0	0,0	92,5	7,5	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	92,5	7,5	6,31
Both cranes down - Store in partner	0,0	59,9	40,1	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,9	59,1	40,1	0,0	0,0	7,31
Both cranes down - Unload spread out	0,0	40,3	59,7	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	40,3	59,7	0,0	0,0	7,25
Both cranes down - Store spread out	100,0	0,0	0,0	0,0	0,0	0,0	0,0	0,2	2,2	37,1	60,5	0,0	0,0	0,0	0,0	7,86

Table D.53: Ranking of policies for all weight combinations with an allowed weight deviation of 2

	Split ranking [%]					Combined ranking [%]										Avg
Policy	1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	**score**
One crane down - 100% capacity	0,0	0,0	0,0	1,4	98,6	0,0	0,0	0,0	1,4	55,5	36,2	6,8	0,1	0,0	0,0	7,90
One crane down - 75% capacity	0,0	0,0	34,7	65,2	0,0	0,0	0,0	34,7	50,8	13,2	1,2	0,0	0,0	0,0	0,0	8,26
One crane down - 67% capacity	2,6	60,5	36,9	0,0	0,0	2,6	60,5	29,1	7,1	0,7	0,0	0,0	0,0	0,0	0,0	8,38
One crane down - 50% capacity	87,0	13,0	0,0	0,0	0,0	86,8	10,9	2,1	0,1	0,0	0,0	0,0	0,0	0,0	0,0	8,50
One crane down - 0% capacity	10,9	26,4	28,4	32,9	1,4	9,8	26,9	29,0	32,9	1,4	0,0	0,0	0,0	0,0	0,0	8,33
Both cranes down - No action	0,0	0,4	3,2	27,2	69,2	0,0	0,0	0,0	0,0	0,0	0,0	0,5	3,2	27,2	69,2	5,66
Both cranes down - Unload at partner	0,0	0,0	5,1	64,1	30,7	0,0	0,0	0,0	0,0	0,0	0,0	0,0	5,1	64,1	30,7	6,25
Both cranes down - Store in partner	0,0	45,9	48,8	5,3	0,0	0,0	0,0	0,1	0,5	0,6	5,6	39,1	48,8	5,3	0,0	7,15
Both cranes down - Unload spread out	0,2	53,5	42,8	3,4	0,0	0,0	0,0	0,0	0,0	0,0	0,1	53,5	42,8	3,4	0,0	7,20
Both cranes down - Store spread out	99,8	0,2	0,0	0,0	0,0	1,4	1,5	4,9	6,7	28,6	56,7	0,2	0,0	0,0	0,0	7,80

Table D.54: Ranking of policies for all weight combinations with an allowed weight deviation of 4

E

Model Manual

This manual will explain how one can use the developed parallel AS/RS Discrete Event Simulation model which takes the upstream and downstream processes into account.

E.1. Input data

The input data should be an Excel file formatted as depicted in figure E.1. The first column should contain the trip/order numbers in ascending order. The second column should contain the release times of orders with a date and time. The third column should contain the departure time of the trip / the time the order needs to leave the output in ascending order. The last column should contain the amounts of goods per sequence in which the trip/order should arrive at the output. For example for trip 0 in figure E.1, 3 pallets belong to the first client, 2 to the second one, 2 to the third one and so on. If the loading order is not relevant, the total amount of goods can be specified as '[26]' for example.

TripID	Earliest start production	Departure time	Pallets
0	2000-12-31 09:45:00	2000-12-31 18:00:00	[3, 2, 2, 3, 2, 3, 2, 3, 2]
1	2000-12-31 09:45:00	2000-12-31 18:00:00	[3, 3, 3, 3, 2, 3, 2, 1, 1]
2	2000-12-31 09:45:00	2000-12-31 18:00:00	[4, 1, 3, 4, 5, 5]
3	2000-12-31 09:45:00	2000-12-31 18:00:00	[3, 2, 3, 2, 2, 3, 3, 2, 1]
4	2000-12-31 09:45:00	2000-12-31 18:30:00	[1, 1, 3, 4, 3, 3, 2, 5]
5	2000-12-31 09:45:00	2000-12-31 18:30:00	[3, 3, 1, 2, 1, 3, 3, 2, 3]
6	2000-12-31 09:45:00	2000-12-31 18:30:00	[1, 2, 2, 4, 3, 2, 4, 3]
7	2000-12-31 09:45:00	2000-12-31 18:30:00	[5, 4, 4, 4, 2, 2]
8	2000-12-31 09:45:00	2000-12-31 19:00:00	[2, 3, 3, 3, 3, 3, 3]
9	2000-12-31 09:45:00	2000-12-31 19:00:00	[5, 1, 1, 1, 5, 4, 1, 2, 1, 1]
10	2000-12-31 09:45:00	2000-12-31 19:00:00	[1, 5, 2, 1, 1, 2, 3, 1, 3, 1]

Figure E.1: Input data format

The name of the input data file should be specified at the lines where 'dataset' is defined. If the system under study requires a different kind of input, for example, if clients should be tracked, the model can be easily adjusted to that.

E.2. Parameters

An overview of the parameters which can be defined, their possible values and an explanation of their influence on the model can be seen in the following tables:

Setting	Possible values	Explanation
Simulation end time	Datetime value	Time until when the simulation runs.
Rack switch-off delay	0-inf seconds	Time it takes after downtime started to switch off a rack. This means no more new trips are appointed to it.
Rack switch-on delay	0-inf seconds	Time it takes after downtime started to turn on a rack.
Production start mode	1,2	Determines production start time for a trip. Mode 1 starts production as soon as possible, mode 2 starts production at a specified offset from departure time.
Production start time	00:00 - 23:59	Time at which production starts during target day.
Last produced departure time	00:00 - 23:59	Departure time of trip on target day until which is produced ahead the day before.
Production offset from departure	0-inf seconds	Offset from departure when production for a trip starts.
Retrieval start mode	1,2	Mode which determines when pallets can move to their outputs. Mode 1 is as soon as possible, mode 2 is from a specified offset from departure.
Output start	0-inf seconds	Time offset before departure when pallets can move to their outputs.
Loading time factor	0-1	Determines how much longer and shorter it takes to load the first and last pallets respectively.
Loading start	0-inf seconds	Time in seconds before departure time when a trip normally starts loading.
Too late threshold	0-inf seconds	Time in seconds before departure time after which a pallet is regarded as placed too late at output.
Max pallets in broken buffer	0-inf	Maximum number of pallets of a trip that can be in the broken buffer for a trip to still be rerouted to another one.
Production pool size	1-inf	Number of trips which can be in production concurrently.
Crane storage time	0-inf seconds	Time it takes to store a broken crane so that the other crane can continue operation.
Crane retrieval time	0-inf seconds	Time it takes to retrieve a repaired crane after storing it.
Downtime start	0-23	Time at which the downtime of the experiment starts.
Downtime duration	0-inf	Duration of crane downtime in hours.
Policy	0-4	Policy to test during experiment.
Experiment	1,2	Experiment 1 is 1 cranes down, 2 is both cranes down.
Run mode	1,2	Determines output generated. 1 = more output, longer computation, 2 = faster, less output.

Table E.1: Possible simulation settings

Setting	Possible values	Explanation
Horizontal crane acceleration	>0 m/s^2	-
Vertical crane acceleration	>0 m/s^2	-
Maximum horizontal crane speed	>0 m/s	-
Maximum vertical crane speed	>0 m/s	-
Base depth pickup time	0-inf seconds	Base time it always takes to pick up a pallet regardless of the depth in which it is placed.
Added depth pickup time	0-inf seconds	Added time it takes per depth rack location to pick up a pallet.

Table E.2: Possible crane settings

Setting	Possible values	Explanation
Input point 1	(x,y,z)	Input point 1 coordinates. Should be within range of rack.
Input point 2	(x,y,z)	Input point 2 coordinates.
Pallet stack input point	(x,y,z)	Empty pallet stack input point coordinates.
Pallet stack output point	(x,y,z)	Empty pallet stack output point coordinates.
Loader division	(a,b) per loader	Outputs to service per loader.
Blocked locations	(x1,y1), (x2,y2), ...	(x,y) coordinates of locations that are blocked and where pallets cannot be stored.
Failure lane	(x,y,z)	Failure lane coordinates.

Table E.3: Possible rack configuration settings

Note that the possible rack configuration settings can be defined for multiple rack configurations.

Setting	Possible values	Explanation
Number of gravity lanes	0-inf	Number of gravity lanes per rack. Should not exceed rack width.
Gravity lane capacity	0-inf pallets	Maximum amount of pallets that can be stored in a gravity lane.
Number of loaders	0-inf	Number of loaders per rack.
Horizontal location distance	>0 m	Horizontal distance between two rack locations in meters.
Vertical location distance	>0 m	Vertical distance between two rack locations in meters.
Pallet stack height	1-inf pallets	Number of empty pallets needed to form one empty pallet stack.
Rack width	1-inf	Number of pallet locations next to each other in the rack.
Rack height	1-inf	Number of pallet locations above each other in the rack.
Rack depth	1-3	Number of pallet locations behind each other in the rack.
Number of racks	1-inf	Number of racks in parallel that form the system.
Number of cranes	1,2	Number of cranes per rack.
Input point capacity	1-inf pallets	Number of pallets that can wait at an input point before blocking the central conveyor.
Crane dwell points	(x,y,x) per crane	Coordinates where the cranes will wait when passive.
Minimum crane distance	0-inf	Minimum number of rack locations in x coordinates that have to be between the cranes to not collide.
Crane service area	(x-x) per crane	Service area of each crane in x coordinates.

Table E.4: Possible general rack settings

The model uses an (x,y,z) coordinate system to define locations with respect to the rack. The x-coordinate represents the width and starts at 0 and goes up to the rack width minus 1. The y-coordinate represents the height and starts at 0 and goes up to the rack height minus 1. The z-coordinate represents the depth and goes from -1 or 1 to the positive or negative rack depth. A negative z-coordinate represents a location on the other side of the two-sided rack. It is assumed that the outputs are placed at the bottom of the rack at locations (x,0,-1). If desired, this could be adjusted in the code.

E.3. Output

The output generated depends on the run mode. If the model is run in run mode 1, an Excel file is generated with extensive outputs featuring an event log, a summary of KPIs, plots for all individual cranes and racks for selected KPIs throughout time and general plots for the whole system. If the model is run in run mode 2, a selection of core KPIs is printed to the output, which decreases computation time. Extra desired KPIs for both run modes can be added by adding Salabim monitors or by collecting data manually in arrays.

E.4. Adjusting the model

The model can be adjusted to other system configurations or operational policies freely by altering the Python code. The code has been accommodated with plenty of comments to help the understanding of the code and making adjustments easier.

E.5. Additional files

Additional files belonging to the model are the input data, which can serve as an example input and be adjusted to one's own system. Next to that, the file 'ExperimentRunner.py' and 'ResultGenerator_auto.py' can be used for automated execution of simulation runs with varying random seeds until the required number of replications to satisfy a confidence interval with a margin of error is reached. This produces Excel files containing the core KPIs split for each crane and rack, and together. To reduce the overall computation time, it is advised to finetune the number of concurrent processes according to the specifications of one's computer.