UNIVERSITY OF TWENTE.

CURING PLATE PROBLEMS

PRODUCTION SCHEDULE EVALUATION FOR ASSESSING EQUIPMENT AVAILABILITY

Author:
Thomas Hugen

Study programme:
BSc Industrial Engineering & Management

Supervisor:
Dr. P.C. Schuur

Preface

Note to the reader: this entire thesis has been anonymized to protect vital information about Company X from competitors.

The report that you are about to read is the final step towards earning my BSc degree in Industrial Engineering & Management. I would like to dedicate this page of the thesis report to express gratitude to the people that helped me through this entire bachelor thesis.

First of all, I would like to thank my thesis supervisor, Peter Schuur, for his support during this bachelor thesis, for his advice, for the fun conversations, and of course, for his patience.

Secondly, I would like to thank Company X for their collaboration. The people at Company X I collaborated with have been very helpful and always made time to answer my questions. In particular, I would like to thank my supervisor at Company X, Wonderful Person, for their guidance in this bachelor thesis, both on a professional and a personal level.

On a more personal note, I would like to thank my family and my friends for their support throughout this bachelor thesis. Specifically, I would like to thank parents. Thank you, mom and dad, for your support throughout this bachelor thesis, your wise counse, and for the chances you have given me throughout my entire life.

Lastly, I would like to thank Paul Kelly for the free online guides they have provided on VBA (www.excelmacromastery.com). Without those guides, I would not have been able to obtain the VBA proficiency that was required for this bachelor thesis.

Management summary

Note to the reader: this entire thesis has been anonymized to protect vital information about Company X from competitors.

This bachelor thesis was conducted at Company X, a factory that produces Product. In the production of Product, Company X uses material handling equipment called MHE, of which Company X uses multiple types. Company X frequently experiences shortages of MHEs in their production. To remedy those recurrent shortages, this bachelor thesis was conducted.

It was identi ed that the MHE shortages were mainly caused by the inability of Company X to consider MHE usage in their weekly production schedule. As such, the following core problem was chosen to be addressed:

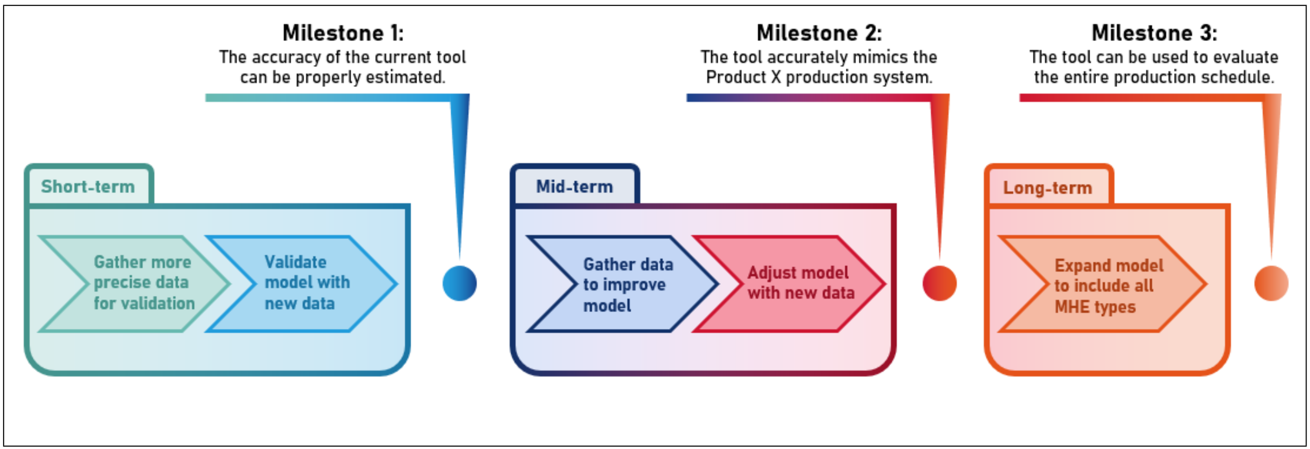
> "the number of MHEs required for running production is not properly considered in the planning of the production process".

To solve this problem, the scope was rst limited to one type of MHE, the "MHE X", that is used to produce a family of Product called the "Product X". To create a pathway towards solving this problem, a literature study was conducted to check what past e orts have been undertaken to solve similar problems. Usual past solutions involved building a Discrete-Event Simulation-based schedule evaluation tool. It was therefore decided to solve the problem in a similar manner.

In order to answer this research question and to develop the schedule evaluation tool, three preparatory steps were conducted: rstly, the production process of pancake wheels was analysed in detail. Secondly, based upon this analysis, a queueing-theory-based conceptual model was developed to serve as a baseline for the Discrete-Event Simulation model. Lastly, to ensure that the schedule evaluation tool aligns with the expectations of the employee of Company X that will use the tool, a short requirements analysis was performed.
With the conceptual model and the requirements analysis in place, we developed a Discrete-Event Simulation model and incorporated in the schedule evaluation tool. The developed schedule evaluation tool is able to generate MHE X usage predictions over any speci ed planning horizon. In doing so, the user can change several settings to match the up-to date production system.

We performed a statistical test to assess the accuracy of the schedule evaluation tool by comparing historical data to simulation data. Though the preliminary test result is that the schedule evaluation

tool is not accurate yet, we are unsure whether this conclusion is valid because the historical data was imprecise. As such, we recommend to gather more precise historical data to derive a nal conclusion. Furthermore, we recommend to improve the model with other data that is to be gathered, as well as expanding the schedule evaluation tool to encompass all MHE types. We propose that the schedule evaluation tool is to be further improved. We present the following road map for further development of the schedule evaluation tool.



Contents

List of acronyms

DES	Discrete-Event Simulation
EDF	Empirical Distribution Function
KPI	Key Performance Indicator
MHE	Material Handling Equipment
MSER	Marginal standard error rule
OEE	Overall Equipment Efficiency
PW1	Process 1 Workstation
PW2	Process 2 Workstation
PW4	Process 4 Workstation
VBA	Visual Basic for Applications
WIP	Work-in-process

Chapter 1

Introduction

This bachelor thesis project was conducted at a factory of Company X, which produces a wide variety and high volume of Products. This rst introductory chapter aims to clarify what the initial motives were for this project, and how this project was to be approach. This chapter starts with a problem identi cation. The second section describes how this problem is to be approached. The third section describes the intended deliverables. The last section provides a general outline of the remainder of this report.

Contents:

1.1. Problem identi cation

In order to provide context to this thesis, we start this chapter by providing with a thorough discussion of the problem at hand. We start this section by stating the management problem that is the underlying reason for this thesis. We continue by identifying the core problem. We end this section by providing a description of our scope.

1.1.1. Management problem

The target availability rate (the percentage of products that are delivered on time in distribution centres and the percentage of products on time in storage for direct transport to customers) of Company X is 97%. Currently, they are only achieving an availability rate of 92%. While there are no signi cant direct costs related to this Availability Rate de cit, management wants to achieve the target level because it bene ts customer relations.

1.1.2. Core problem

Management has indicated that the main cause for production delays are recurrent Material Handling Equipment (MHE) shortages at Process 1. These MHE ... are used to aid the production at Process 2. Secondly, they carry the Products and keep them separate whenever these products are Batched to prevent sticking. A picture of such an MHE can be found in Figure 1.1.

Classified

Figure 1.1: Classi ed picture

At times, no MHEs are available at Process 1 because they are all being used, requiring Process 1 to be stopped until a set of MHEs becomes available again. It is estimated that these shortages historically [cause signi cant problems]. At present, shortages do not occur anymore because a considerate batch of new MHEs have been bought, but management indicated that in the past, shortages returned within a couple of weeks after restocking.
To nd the root causes for these shortages, the problem was analyzed by asking several problem owners about their thoughts on the underlying problems and by observation. This analysis [is classi ed]. Do note that the diagram only depicts all problems related to MHEs; other problems were observed as well, but they were not considered for this project as MHE shortages have been identi ed as the main culprits for production delays. In the end, it was decided with Company X's management that the problem to be solved should be:

> "the number of MHEs required for running production is not properly considered in the planning of the production process".

It was chosen because it ts within the scope of a bachelor thesis while solving it is expected to be most impactful on MHE availability.

1.1.3. Scope

Since Company X produces a plethora of different Product types, they also use different types of MHEs. To keep this research manageable, this bachelor thesis focuses on availability of one type of MHE: the "MHE X". The MHE X was chosen because there is more knowledge available about this MHE type, and because the MHE X is used to produce about 50% of the total production volume of Company X. Including other Product types would require a vast amount of additional work due to the differences in production processes and knowledge gaps regarding the other MHE types.

1.2. Problem solving approach

To form a pathway to solving the presented problem, we first provide an analysis of the problem, followed by a description of literature on similar problems. Combining the analysis and the literature, we identify the knowledge gaps that need to be filled before the problem can be solved. Based on those knowledge gaps, we end this section with a formulation of the research questions of this thesis.

1.2.1. Problem analysis

Conversations with several staff members confirmed that this problem is not omnipresent; only on certain moments the number of MHE Xs that is being used spikes, resulting in the aforementioned shortages at Process 1. As such, the problem could be very well solved by evaluating schedules in order to ensure that MHE X requirements are divided evenly over the schedule's horizon.
However, with the current configuration of Company X's production scheduling, this is a hard task: at present, only Process 1 is being scheduled, and the other production processes are managed by the factory's labor staff. Because the other production processes are not being planned, the flow of a MHE X and its return to Process 1 is unpredictable. This disables the possibility of evaluating whether a candidate schedule has an even division of MHE X requirements over its horizon.
An attempt to solve this problem in this manner has been made in the past; an estimation of the cycle time of MHE Xs has been developed earlier in an attempt to evaluate schedule performance with regards to MHE X availability. However, this cycle time estimation was based on a constant throughput and constant process times. This is an unrealistic assumptions given the process variability of the manufacturing processes at hand.
Therefore, a solution to this problem should provide insights into the flow of MHE Xs, taking into account process variability, such that candidate schedules can be evaluated based on their predicted MHE X requirements over time. Based on these predictions, it can be decided whether the candidate schedule needs to be altered. As this has to be done repeatedly, some sort of tool should be developed that can be used to perform this evaluation.

1.2.2. Literature

To gain insight in how similar problems have been solved before, a literature review was conducted within the subject of schedule evaluation. Three main approaches have been found.
Firstly, Tardif & Spearman [1] propose \a procedure designed to detect and remedy scheduling in-

feasibilities" with an MRP-based approach. The approach estimates Work-in-process (WIP) within a system over a certain schedule horizon, which is used to evaluate whether there is enough WIP within the system to meet daily demands. A aw of this approach, however, is that it assumes deterministic cycle times.
Secondly, Haro et al. [2] present a model of a manufacturing process that attempts to predict schedule infeasibilities based on a stochastic demand/supply analysis. However, the supply rate of MHE Xs within Company X's production system is largely unknown because, as mentioned in the previous section, the cycle time of each MHE X is unknown. The use of this method would ironically require another tool to estimate these cycle times.
Lastly, Four research projects ([3], [4]; [5] and [6]) utilize Discrete-Event Simulation (DES) to evaluate production schedule performance. A major advantage of DES over the other methodologies is that it can account for the variability within production processes [3].
Based on the descriptions, DES seems to be the most promising methodology for developing a tool that could solve the problem at hand. As such, we decided to develop a schedule evaluation tool using DES.

1.2.3. Knowledge gaps

In order to build such a schedule evaluation tool, a number of things have to be investigated. Firstly, the manufacturing system of Company X needs to be analyzed in detail in order to create the DES-model. Speci cally, the production routing, Process times and queue behaviours need to be modelled. Secondly, it should be de ned how the the tool exactly should function by gathering the needs and wishes of the stakeholder that will eventually be using the tool (the Problem Owner). Lastly, it should be investigated how the DES-model should be developed, i.e. which type of model to use and what software to use.

1.2.4. Research questions

In order to resolve the aforementioned knowledge gaps, the following research question has been formulated:

> "What does a DES-based tool that predicts MHE X requirements resulting from a candidate production schedule of Company X look like?"

To answer this question, several subquestions have been formulated that, answered in this order, should guide towards an answer to the main research question.

SQ 1: What processes are involved in the production of Product X at Company X?

SQ 2: How should Company X's manufacturing process for the creation of Product X be modelled?

SQ 3: What are the needs and wishes of the Problem Owner regarding the functioning of a schedule evaluation tool?

SQ 4: How should the model of Company X's manufacturing process be translated to a DES-model?

1.3. Deliverables

Naturally, the aforementioned DES-model has been delivered. Along with the model, a report describing the model and its development process (which is this report) has been delivered as well.

1.4. Development approach

The research questions formulated in the previous section has been answered in several steps. The initial step, related to research question 1, is to obtain a detailed description of the production process of Product X at Company X. This question is answered in Chapter 2. To answer research question 2, the description provided in Chapter 2 was translated into a conceptual model that is described in 3. Chapter 4 aims to provide an answer to research question 3 by summarising how the tool should be developed. Afterwards, to answer research question 4, Chapter 5 describes how the conceptual model is translated into a DES model.
Unrelated to our research questions, there are two more chapters: Chapter 6 aims to describe to what degree the DES model results are valid. Finally, in Chapter 7, we reflect upon the creation process and the results, and we provide recommendations for future improvement.

Chapter 2

Company X's production process

The aim of this chapter is to answer our rst research question:

> "What processes are involved in the production of Product Xs at Company X?"

In order to provide an answer to this question, numerous informal interviews were conducted with employees of Company X. From these interviews, we developed a description of all relevant processes for the production of Product Xs, is discussed in this chapter. The chapter starts with a high-level process description, followed by a section containing more information about MHEs. Afterwards, four sub-processes involved in the production are discussed separately. The last section discusses the planning e orts at Company X.

Contents:

2.1. Process description

From a high-level perspective, the production of Product Xs consists of ve consecutive steps:

1. Process 0
2. Process 1
3. Process 2
4. Process 3

5. Process 4

The production process is visualised in Figure 2.1. For this thesis project however, the mixing process bears no relevance because MHEs are not being used in this part of the production process, and thus, Process 0 has been left out of the discussion.

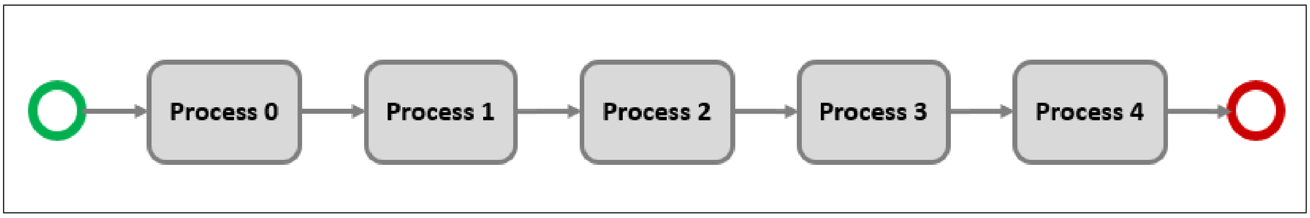


Figure 2.1: High-level schematic overview of the production process of Product X.

Each of these separate processes has several workstations. Products go through each of the four processes and need to visit only one workstation per process. It should be noted that a Product cannot be produced with every machine; each Product. Each Process 1 Workstation (PW1) is specialised to produce only a speci c set of Product types. Similarly, while each Product can be processed at every Process 2 Workstation (PW2), the Product requires a speci c Process 2 setting. Likewise, each Process 4 Workstation (PW4) can only process a speci c set of Product types. As such, each Product has its own routing.

Which Products are being produced depends on the production orders that have been scheduled. The schedule speci es for each order on which PW1 it is scheduled and the designated oven program. However, in which oven Products will be cured and the PW4 at which those Products will be unpacked is not speci ed on beforehand, and is managed by the sta on the spot.

In the following six subsections, more information will be given about the MHEs, about each relevant process and about the planning process of the factory.

2.2. MHEs

.... Process 2 requires batching due to the long process times, but if the Products batched, they might stick to each other during Process 2. The MHEs solve that problem by keeping the Products separate. The MHEs are reusable; after the Process 4, MHEs are returned to Process 1 for reuse.

There are several di erent types of MHEs. For comprehensiveness, it is de ned that MHEs speci cally used for Product Xs are called "MHE Xs", and that an MHE that carries Products is called a "loaded" MHE. For material movement and process e ciency, the loaded MHEs are by batching them in "a Batch" during Process 1, until they are nally unbatched at Process 4. A number of Product Xs t on one MHE X depending on the size of Product X. For a comprehensive visualisation of this terminology, see Figure 2.2

It is estimated that around 215 MHE Xs t in one Batch if they are loaded with Products, and 400 MHE Xs t in one Batch if they are unloaded. These numbers are variable, but the variable component cannot be accurately estimated due to a lack of data.

The transportation of Batches between workstations can happen up to two at a time.

Figure 2.2: Classified picture

2.3. Process 1

There are a number of PW1s involved in the production of MHE Xs (out of 19), which are categorised in Table 2.3. The back-up PW1s, as mentioned in Table 2.3, are used in case an order needs to be processed but no other Product X PW1s are available

Figure 2.3: Classified table

The PW1s both process the Products and load them on MHE Xs. Once fully loaded, the MHE X is batched. If the Batch is full, the Batch will be finished. Note that this finishing process, which takes no more than 10 minutes, is done separately from the PW1s; while a Batch is being finished, another Batch can be processed at Process 1.

Process 1 deals with daily order delays that on average last between 10 and 60 minutes. These delays can be caused by numerous factors, such as machine malfunctions or lack of materials. When the delay occurs and its resulting waiting time is unpredictable.

2.4. Process 2

Whenever a Batch is nished at Process 1, it is transported to Process 2 and immediately queued at the PW2 with the Process 2 setting that is scheduled to be ignited next. These PW2 are chambers that can t up to [speci ed number] Batches. There are [speci ed number] PW2s that each can run the all di erent Process 2 settings, which have been summarized in Table 2.4. Note that not only the Product Xs use the three involved Process 2 settings; PW2s often contain a mix of Batches and Batches with other Product types. These other Batches also have a di erent size, which in uences the e ective capacity of a PW2.

Classified

Figure 2.4: Classi ed table

While there are X PW2s in theory, usually, a number of PW2s is broken or unavailable. Thus, while the Process 2 capacity is theoretically enough, queues can form in case a large percentage of PW2s is unavailable. It should be noted that Product Xs have priority over the other Products when it comes processing, so in case of queueing, Product Xs take precedence.

To ensure limited ow variability for MHE Xs, approximately every [speci ed time period]s an PW2 is scheduled with Process 2 setting 3. This means that a PW2 with Process 2 setting 3 will start either [speci ed time period] after the previous PW2 with Process 2 setting 3 was started or when the queue is long enough.

After Process 2, the Batches will be transported to Process 3.

2.5. Process 3

Company X has [speci ed number] PW3s in which Batches are processed. At this process, there is no distinction between Product types; nearly all Products are processed in Process 3. The PW3s ....
The PW3s have approximately the same capacity as a PW2s. Whenever Batches are transported

from Process 2 to Process 3, the Batches are divided over the PW3s. Occasionally, capacity has been reached at the PW3s, which causes queuing of Batches. However, Product X Batches take precedence over other Batches.
After the designated process time has passed, the Batch will be transported to Process 4.

2.6. Process 4

The PW4s involved in the production of MHE Xs are summarized in Table 2.5. [One PW4 is manual, and the other PW4s are semi-automated, meaning that a machine is doing work and an employee is doing work].

Figure 2.5: Classi ed table

The semi-automated and automated PW4 each process up to two Batches of the same order before requiring a setup of two new Batches. This setup takes approximately [speci ed time period]. During this setup, the employee working at the PW4 can continue processing the Products. Between Batches of di erent orders, the PW4s requires a more complex setup that can vary in time because it depends on how fast the employee can keep up. It is expected to take [speci ed time period]. The oor sta determines which Batches are processed next based upon a least amount of setups logic, with earliest due date being the secondary priority.
The process times of Process 4 are highly variable, and are largely dependent on four factors. Firstly, there is a human-factor; each employee working at the lines has a di erent processing speed. Secondly, the operation to be performed can di er, and some operations take longer than others. Thirdly, the processing speed between PW4s di er. Lastly, errors, either human or non-human, in uence the process. The PW4s, the Products sometimes require adjustment, and the employee can make mistakes that decrease the processing speed. The PW4s also break down sometimes, but these breakdowns are uncommon.
There are a plethora of operation types for Process 4 at Company X, but in this bachelor thesis, we focus on the three main operation types and place the other operations in one category. See Table 2.1

for a summary of these operations.
At Process 4, the products are separated from the MHE Xs. The empty MHE Xs are collected in a Batch that will be returned to Process 1 when two of such Batches are full. In case Process 1 is short on MHE Xs, they are transported earlier.

Abbreviation	Operations type
"OP1"	Operation 1
"OP2"	Operation 2
"OP3"	Operation 3
"OTH"	All other operations

Table 2.1: Summary of the di erent operations at Process 4.

2.7. Planning

The entire plant is operative Operations 3 a day, 5 days a week. The production week starts on Sunday evening, and ends on Friday evening. Process 2 can be run during the weekend without requiring assistance from employees, and as such, at the end of the week, all Batches that are being processed at Process 1 will be nished and put in Process 2, even if those Batches are not full. All PW2s will run during the weekend if they have batches to process. On Sunday evening, all of these Batches can directly be transported to Process 4 because Process 3 can be skipped if Process 2 runs during the weekends.
The production schedules are created by the Problem Owner in their ERP system. Orders are scheduled to follow a "least amount of setups" and "earliest due date" logic; orders that approach the due date are scheduled rst. All orders that do not require Process 1 setups due to similarities with the previous order are preferably scheduled afterwards. Whether setups are requires is also speci ed in the production schedule.

Chapter 3

Conceptual model

This chapter aims to provide an answer to our second research question:

> "How should Company X's manufacturing process for the creation of Product X be modelled?

To answer this question, we first developed a theoretical framework for the model itself. Building on that framework, we used the information from the previous chapter to build our conceptual model.
It should be noted that the production system of Company X was too complicated to model analytically. Therefore, our conceptual model in this chapter is described such that it can be directly translated to a DES-model, although we do lend some queueing theory to derive the model.
This chapter starts with a short resume of theory and definitions to enable the communication of our conceptual model. The concurrent section describes the model from a high-level perspective. In the last section, we dive into the details of our conceptual model.

Contents:

3.1. Theory and definitions

To describe the conceptual model, we introduce some notation along with a brief description of theory. Firstly, we discuss our notation for the applied queueing theory. Afterwards, we discuss some theory surrounding empirical distribution functions.

3.1.1. Queueing theory

The conceptual model we created is primarily based on queueing theory. We introduce the following terminology:

Customer: A specific type of object that flows through the modeled system.
Server: An activity that interacts with the customer.
Network: A group of multiple servers that are connected in tandem and/or in a parallel manner.

Each server has an arrival process, dictating the number of customers arriving at that server over a specific period of time, and a service process, dictating the number of customers that are being processed at that process over that same period of time. Note that in case of networks, the arrival process of a certain process is governed by it's predecessor.
Furthermore, each server has an arrival process with a corresponding interarrival time ^{1}, which is the number of customers or batches of customers arriving at that server over a specific amount of time, and a service process with a corresponding service time ^{1}, which is number of customers or batches of customers that interact with that server over a specific amount of time.
Within the field of Operations Management, servers are known as "stations" and service times are better known as "cycle times" (defined by Hopp & Spearman, 2008, as: \average time from release of a job at the beginning of the routing until it reaches an inventory point at the end of the routing"). According to Hopp & Spearman [7], the cycle time of a single station consists of the following components:

Move time: the time jobs spend being moved from the previous workstation;
Queue time: the time jobs spend waiting for processing at the station or to be moved to the next station;
Setup time: the time a job spends waiting for the station to be set up;
Process time: the time jobs are actually being worked on at the station;
Wait-to-batch time: the time jobs spend waiting to form a batch for either (simultaneous) processing or moving;
Wait-in-batch time: the average time a part spends in a (process) batch waiting its turn on a machine and
Wait-to-match time: the time a component spends waiting for their mates to allow the assembly process to occur.

These cycle time components are used to model the total service times of each server. Note that not all of these cycle time components are always nonzero.

3.1.2. Empirical distribution functions

During modelling, we attempted to mimic certain stochastic processes (such as service times) by trying to derive the underlying probability function. In doing so, we collected historical data on those processes and attempted to fit certain probability distributions over the data. Afterwards, we assessed the fitness of those probability distributions using a Chi-squared test. However, none of these attempts passed the Chi-squared test, and as such, we decided to mimic each of those stochastic processes using an Empirical Distribution Function (EDF) instead.

We define an EDF to be a cumulative distribution function of a sample. To clarify, let X_1, X_2, ..., X_m be an ordered sample of data points. Then, F(X) (the EDF) returns the probability of X or smaller, i.e. the proportion of elements X_i in that sample smaller than X. For clarification, we refer to Figure 3.1 where this concept has been illustrated.

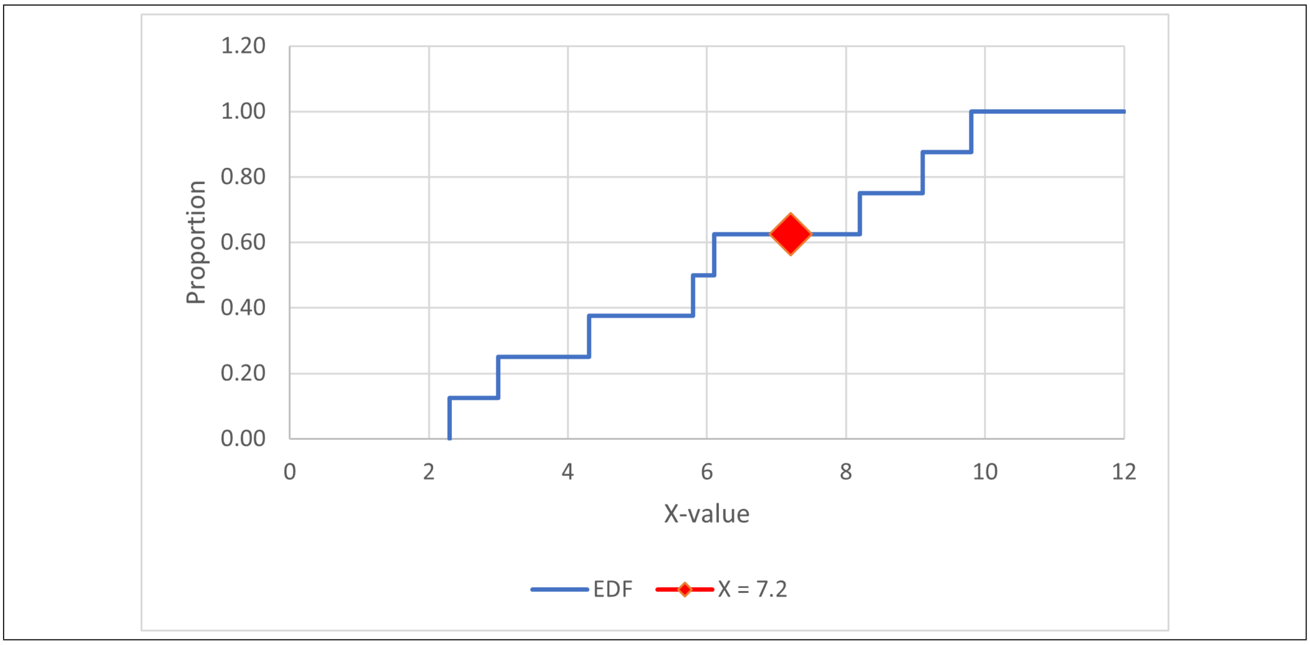


Figure 3.1: EDF of the ordered sample $S = \{2.3, 3, 4.3, 5.8, 6.1, 8.2, 9.1, 9.8\}$. The red dot represents the value $X = 7.2$. The reader can verify that the proportion of observations in S smaller than X is 0.625.

In order to mimic a stochastic process, we are interested in a function that generates values representative of the underlying process given a certain random variable $r \in (0, 1)$. As such, we are mainly interested in the inverse empirical distribution function, denoted as $F^{-1}(r)$.

For the construction of these inverse empirical distribution functions, we used a methodology similar to Bratley, Fox and Schrage [8]; they advocate to model the right tail of the empirical distribution function as an exponential function with shape parameter to allow drawing larger values than previously observed. Furthermore, for large sample sizes, values for the EDFs are usually divided over bins of equal size to reduce computation time. Furthermore, for more accurate representation of a stochastic process, the space between bins is often treated as a linear function to encourage more continuous behaviour of the EDF.

With these modelling descriptions in mind, we have constructed our inverse EDFs as follows:

Let X_1, X_2, ..., X_m be an ordered sample of a certain stochastic process, and let that sample be divided over $n = ceil(\sqrt{m})$ bins. Then, our inverse empirical distribution function $F^{-1}(r)$ is given by:

$$F^{-1}(r) = \begin{cases} B_i + (B_{i+1} - B_i)\frac{r - p_i}{p_{i+1} - p_i} & \text{for } p_i \leq r < p_{i+1} \\ B_n - \theta \ln\left(\frac{1-r}{1-p_n}\right) & \text{for } r \geq p_n \end{cases} \tag{3.1}$$

With:

r : a number pseudo-randomly drawn from the range [0,1)

p_i : The lower bound probability of the ith bin, with $p_0 = 0$

B_i : The lower bound data point of the ith bin, with $p_0 = X_0$

$i = 0, 1, ..., n$

θ : A shape parameter, given by:

$$\theta = \frac{X_{m-k} + \sum_{l=m-k+1}^{m}(X_l - X_{m-k})}{k}$$

To clarify this equation, we provided an example in Figure 3.2 for a graphical reference. A difference with the methodology by Bratley, Fox and Schrage [8] is that they do not divide the data over bins. Since our data sets typically contain more than 500 data points per sample, we decided to use bins to decrease computational time significantly for a minimal sacrifice of accuracy.

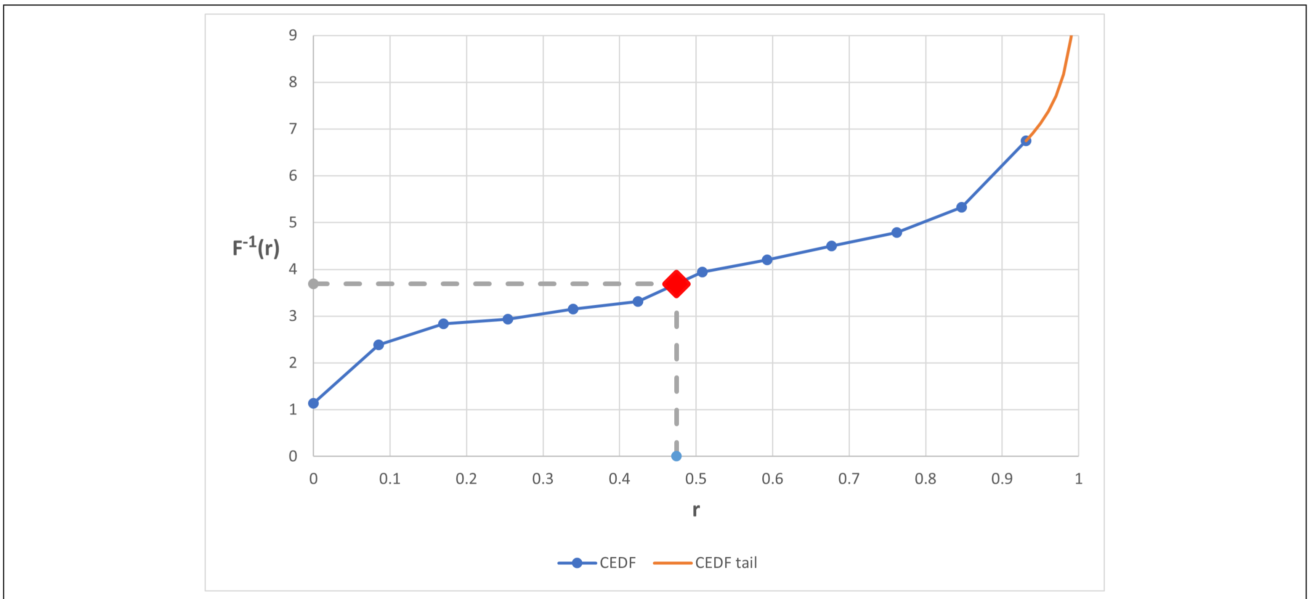


Figure 3.2: Inverse EDF of a fictive sample using the methodology of Bratley, Fox and Schrage [8]. The red diamond represents the value $r = 0.47$. r lies between the points ($p_6 = 0.424$; $B_6 = 3.31$) and ($p_7 = 0.508$; $B_6 = 3.94$). As such, value corresponding to r is $F^{-1}(r) \approx 3.89$. The orange line to the right of the figure represents the tail of the EDF that has been approximated with an exponential distribution.

3.2. High-level model description

Given the presented theory, we sketch a rough outline of the model in this section. We firstly discuss the envisioned customers and the network and flow. Furthermore, the arrival processes and the service processes are briefly introduced; a more in-depth discussion of the model can be found in Section 3.3

3.2.1. Customers

The DES-model should estimate MHE X usage resulting from a candidate production schedule. Since MHE Xs carry Product X, MHE X usage can be determined directly through estimating the amount of work-in-process (WIP) present in the system. WIP flow should therefore be tracked to estimate the MHE X usage resulting from a candidate production schedule. As WIP spends the entire cycle time between Process 1 and the unstacking in Batches, the easiest way to analyze WIP flow is to track the flow of Batches within the system. Thus, the Batches are the entities for this model.

If this model is to mirror reality perfectly, there should be a finite customer population because there are only a finite number of MHE Xs in stock and therefore also a finite number of Batches that can be present in the system. However, we simplified this model intentionally to have an infinite customer population. The reason for this assumption is that shortages at Process 1, which happen in real life,are very undesirable to include in the model; picture a situation where those shortages are included in the tool, and the tool predicts three shortages. In this case, it is unknown whether the two latest predicted shortages are an indirect result of the first shortage or if they are also caused by overscheduling. This means that the user of this tool would have to try to remedy each shortage separately in order to generate a better schedule.
Instead of including shortages in the model, we intend to create the model such that it shows the MHE X usage at any point in time. The tool shows if this usage exceeds a threshold but this excess does not result in a shortage within the model itself, such that multiple predicted shortages can be remedied all at once.

3.2.2. Network and flow

As discussed in Section 2.1, four different types of activities are relevant for this bachelor thesis. Each of these activities has a number of servers that together form a network. As described in Section 2.1, there are [specified number] of PW1s, [specified number] of PW2s, [specified number] of PW3s and [specified number] of PW4s involved in the creation of Product X. Since MHE X shortages are excluded from the model, the production process can be simplified to a linear model. Given these modelling decisions, we derived the queueing network depicted in Figure 3.3.

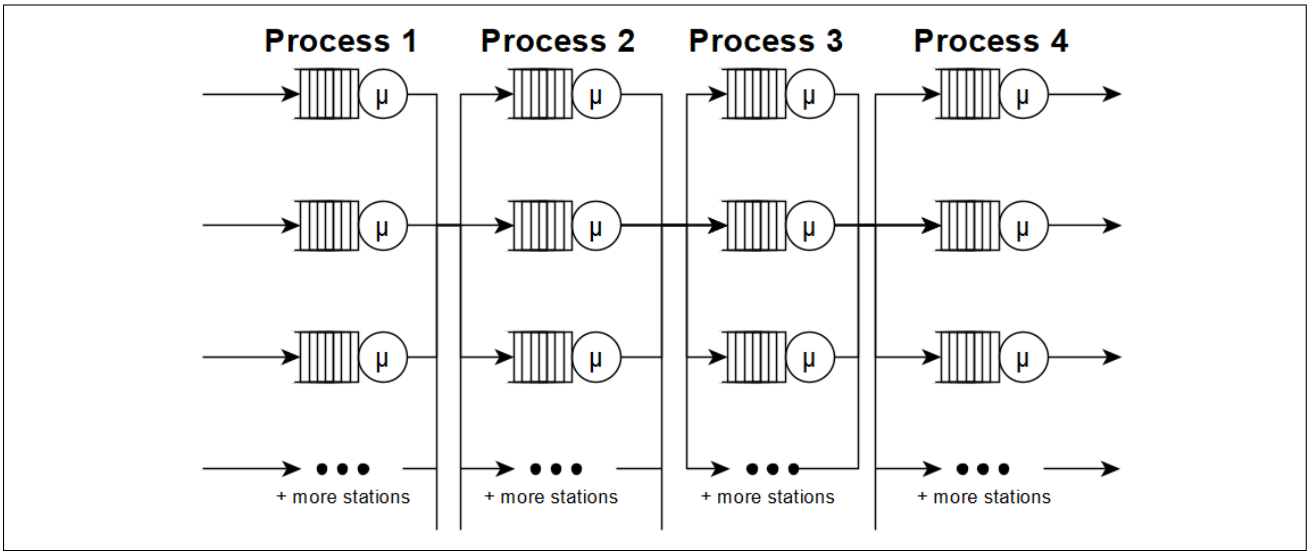


Figure 3.3: (Classi ed) depicition of the queueing network

The ow of Batches through this network, which is modelled to mirror reality as accurately as possible Batches, is comprised of the following steps, in which one single Batch is followed:

1. The Batch is processed individually at Process 1
2. The Batch is batched at a PW2 with other Batches before Process 2 starts.
3. The batch of Batches undergoes Process 2.
4. The batch of Batches is transported to Process 3 and undergoes the process.
5. The batch of Batches are transported to the queue at Process 4 and are separated.
6. The Batch is processed at Process 4.
7. The Batch leaves the system.

3.2.3. Arrival processes

Within tandem queues, the arrival processes of all servers are governed by the service rate of their predecessors, except if those servers have arrivals from outside the system (such as servers at the start of the system). As such, for Process 2, Process 3 and Process 4, all arrival processes depend on the service rate of the predecessing process. The arrival process at Process 1 depends on when the orders are scheduled within a candidate production schedule.
One exception in which the arrival processes of the servers is independent of their predecessors is on Fridays and Sundays: the factory is closed during the weekend, and on Friday, all WIP present in at Process 1 will be transported to Process 2. On Sunday evening, when the factory starts operating again, all Process 2 settings have nished and Process 3 will be skipped, so those Batches will be queued up for the Process 4 instead.

3.2.4. Service processes

The cycle time of each process results from how those process are being modelled. In the next section, we describe how we modelled each process, and thus this section also provides with a description of how we modelled the cycle time of each process. Note that matching does not takes place within the production process of Company X, and therefore, the wait-to-match time is 0 for all modelled processes.

3.3. Process models

This section describes how Process 1, Process 2, Process 3 and Process 4 have been modelled. For each of these processes, at least a general process description and a description of the service process is given. For the discussion of these service processes, we use the di erent cycle time components discussed in Section 3.1.1.
As stated in Section 3.2.3, the arrival process of servers in networks is dependent on the service process of their predecessor, only the arrival process of Process 1 is relevant to discuss. Estimated queue times are not be discussed; these are too complicated to model analytically. The queue times are included in the simulation model and arise from running the model.
To provide some more contextual information, this section starts with a description of general modelling assumptions.

3.3.1. General modelling assumptions

To clarify the upcoming sections, we start by elaborating on three aspects on the model for which we made some assumptions: the model initialization, what happens if a Batch is nished, and irregularities in time progression. After these general modelling decisions have been speci ed, we continue with discussing the model of each server.

Model initialization

The production schedule doesn't specify individual Batches in the Process 1 planning, but it speci- es orders. As such, the model is initialized by splitting all orders into Batches. As has been discussed in Section 2.2, it is estimated that around [speci ed number] loaded MHE Xs t in one Batch. There is no data available on variability of this number, so it is be taken constant. If an order does not have an order size that is a multitude of [speci ed number], the remainder is batched into the last Batch. Usually, the number of Product X per MHE X is [speci ed number], but this is not always the case. It is taken in our conceptual model as an input variable.

End of the network

MHE Xs follow the same ow as the Batch they are assigned to except for when the Batches are nished at Process 4; at Process 4, all empty MHE Xs are collected and divided over two Batches. If these Batches are full, the empty MHE Xs are transported to Process 1 with a move time that is assumed to be a constant [speci ed time period]. We expect that it is substantially less in the Company

X's production system. As discussed in Section 2.6, it is estimated that around [speci ed number] empty MHE Xs t into one Batch. Again, no data is available on the variability of this number, so we assume the Batch-capacity for MHE Xs to be that number and constant. With these assumptions in mind, we model the return of MHE Xs from Process 4 to Process 1 to happen immediately once two Batches full with MHE Xs have been reached at Process 4.

Time progression

As within the production system of Company X, our model has three shifts: Shift 1, Shift 2 and Shift 3. The relevance of the inclusion of these shifts is discussed in Section 3.3.5. With the inclusion of the shifts, we model the weekend to start half an hour before the end of Shift 2 on Friday, and end at the start of Shift 3 on Sunday; before the weekend starts, the machinery needs to be shut down, which is assumed by Company X's production leader to take [speci ed time period].

3.3.2. Process 1 model

The high-level Process 1 process for one Batch is schematically represented in Figure 3.4.

Figure 3.4: Classi ed model.

Our Process 1 model features the following simpli cations in comparison to reality:

- In Company X's production system, a Product X is created, then loaded in an MHE X, and if that MHE X is full, it is added to a Batch, and this process is repeated. To reduce the number of computations, we model the process such that rst all Product X are created, then loaded on MHE Xs, and then added to the Batch.

- Since we do not have sufficient data on the machine malfunctions that occasionally happen at the PW1s, these events have been excluded from the model. A correction for this exclusion is discussed at the service process discussion (under Section 3.3.2: "Service process").
- To reduce the number of computations within the simulation model, we introduce the simplification that all MHE Xs required for the processing of a Batch are all reserved immediately at the start (instead of the real-life situation, where the MHE Xs are added gradually). This does not have any direct implications on the model itself.

Arrival process

The arrival process for Process 1 is dictated by the input production schedule. As a simplification, we model this arrival process such that an order is added to the queue on the day it starts (i.e. the number of Batches required for the processing of that order arrive all at the same time on the day the order needs to be processed, at 00:00). This simplification has the direct implications that an order cannot be started earlier than scheduled, and that a PW1 might become idle for some time if there are no orders left in the queue.

Service process

Based on our model, the only cycle time components that play a role are the setup time and the process time. The queue time and move time are set to zero because MHE Xs are in use only after Process 1 have started. No batching of Batches takes place at Process 1, so wait-to-batch time and wait-in-batch time time are irrelevant.
Occasionally when a new order is being processed at Process 1, that order requires a setup. The setups and their corresponding times are specified in a production schedule, and thus setup times can easily be modelled by adding the specified setup time to the cycle time of the first Batch of the new order.
The process time of a Batch can be calculated by multiplying the takt time of a PW1 (t_T, the theoretical time between the output of one Product X and the next) with the amount of Product X that need to be processed for that Batch. Though the takt time of a PW1 is not constant (see Section 2.3), there is limited data on their occurrence and the resulting waiting times at the time of writing, and thus, the variability of the takt time cannot be modelled reliably. However, these waiting times can be modelled through the Overall Equipment Efficiency (OEE), which is at Company X the total valuable operating time of a machine divided by its total operating time, excluding setups; using the OEE, an "effective takt time" (t_{Te}, obtained by dividing the takt time by the OEE) can be calculated to represent the process time. A limitation of this method is that the OEE is an average, and thus does not capture process time variability.
For each PW1, the theoretical takt time, the OEE and the effective takt time are shown in Table 3.5.

Given these cycle time component descriptions, the cycle time of a specific Batch i at a specific PW1 ($_{P1}^{1}(i;PW1)$) is given by:

$$_{P1}^{1}(i;PW1) = t_{Te}(PW1) \quad Q(i) + t_S(i)$$

Classified

Figure 3.5: Classified table.

With:

$$
\begin{aligned}
t_{Te}(PW1) :&\ \text{The effective takt time of a certain PW1.}\\
Q(i) :&\ \text{The quantity of Product X in Batch } i\\
t_S(i) :&\ \text{The setup time required for starting Batch } i
\end{aligned}
$$

3.3.3. Process 2 model

Modelling Process 2 is challenging due to the interference of other material streams with the PW2s involved in the creation of Product X. This causes the following two uncertainties:

1. As mentioned in Section 2.4, some other Product types are processed alongside Product X in the same PW2, which reduces the effective PW2 capacity. How many Batches of other Product types will be included in a PW2 is unknown in advance.
2. Since all PW2s can run all Settings and some PW2s are needed for processing other MHE types, it is uncertain how many PW2s are available for processing Product X.

These two phenomena significantly increase model complexity; for them to be represented perfectly, all other material streams will have to be included in the model as well. Since this is outside the scope of this bachelor thesis, we use different strategies to address these challenges, we introduce the following simplifications:

- We set a maximum for the number of Batches allowed for each Process 2 setting. Furthermore, we introduce an inter-start time: the time between PW2 process starts of the same Process 2 setting. This inter-start time is pseudo-randomly sampled from a distribution that has been derived from historical data. The inter-start time allows the simulation to mimic the real system, while the maximum ensures that, when a large inter-start time is sampled, it does not result in unrealistic system behaviour (e.g.: a PW2 starts with [unrealistically large number] of Batches in it). We discuss this in more detail in 3.3.3.

- To address the second challenge, we simplify the model by using the logic that Product X take precedence over other Product types (from Section 2.4) by assuming that, if the Process 2 capacity has almost been reached, a PW2 for Product X will be made available instantaneously to avoid queueing for Product X. This simpli cation is justi able by the fact that smooth ow for Product X is a priority for the factory oor sta . With this simpli cation, the Process 2 for Product X can be modelled such that the possibility of queueing is eliminated.

Now, with these two simpli cations in mind, we model the process as follows:

Figure 3.6: Classi ed model.

Inter-start times

To model the inter-start times, historic data on the PW2 starts at Company X between [speci ed time period] was gathered and analyzed. This time period was chosen because substantial amount of data was required to model the inter-start times, and the Process 2 settings have not changed signi cantly over that time period.

We initially attempted to t statistical distribution functions to describe the distribution of the historic data. For each of those attempts, the tness of those statistical distributions was tested using a Chi-square test. Unfortunately, after several attempts, we did not manage to find a statistical distribution to describe the inter-start times with sufficient accuracy to pass a chi-square test. As such, the inter-start times of each Process 2 setting have been described using continuous empirical distribution functions based on the historical data, denoted by $F_{P2}^{-1}(r)$, where O represents the Process 2 setting.

The presented methodology does have one limitation: while $F_{P2}^{-1}(r)$ can be used to mimic the behaviour of the inter-start times, the case could also be in our model that a PW2 starts when the maximum capacity has been reached before the inter-start time has been reached. This means that our model predictions for the inter-start times are on average more optimistic than in reality. To balance this, $F_{P2}^{-1}(r)$ should only be constructed from a sample of inter-start times when PW2s were not full, however, the required data for this was unavailable.
Now, let t_f be the predicted time that will elapse before a PW2 reaches capacity, then the inter-start time $IST(r;t_f)$ of that workstation is given by:

$$IST_y(r;t_f) = \min\left(F_y^{-1}(r);t_f\right) \tag{3.2}$$

Service process

With the current model in mind, the relevant cycle time components are the move time, process time and the wait-to-batch time; we eliminated the queue time by assuming that there is always a PW2 available for MHE Xs, the PW2s do not require setups, and there is no wait-in-batch time because all Batches are processed simultaneously.
The move time, which is the time a Batch spends being moved from the PW1 to Process 2, is assumed to be [specified time period]. The finishing of a Batch at Process 1 (as described in Section 2.3) is included in the move time, because the finishing of the Batch happens separately from Process 1 and is done halfway the moving of a Batch.
The process time is modelled to be constant, since the Process 2 settings at Company X are run for a fixed time. Table 3.7 shows these process times corresponding to each Process 2 setting.
The wait-to-batch time is given by equation 3.2).
With this model in mind, the Process 2 cycle time ($_{P2}^{1}(i;PW2;y)$ of a certain PW2 is given by the following equation:

$$_{P2}^{1}(i;PW2;y) = t_{M(PW1->PW2)} + IST_y(r;t_f) + t_P(PW2)$$

With:

$t_{M(P->PW2)}$:	The move time between a PW1 and a PW2, assumed to be 10 minutes
$IST_y(r;t_f)$:	The inter-start time represented in Equation 3.2
$t_P(y)$:	The process time of Process 2 setting y

Figure 3.7: Classified table.

3.3.4. Process 3 model

Just like Process 2, other Product types utilize the PW3s as well, which in a similar fashion creates the uncertainty whether queueing at Process 3 will occur because we do not know anything about the material streams of other MHE types. However, according to our data, the probability that a Batch arrives at Process 3 while all of the PW3s have reached capacity is estimated to be less than 3% (see Appendix B.1), and again, in case queueing is expected to happen, Product X take precedence over other Product types. As such, the same methodology as with the Process 2 is applied here; to simplify the model, Process 2 is also modelled as a process where the number of servers does not matter and the possibility of queueing is eliminated.
Given this modelling paradigm, we model of Process 3 as follows:

Service process

A first important consideration for modelling the Process 3 service time is that a Batch skips this step if it was transported to Process 3 at the end of the week, rendering the total Process 3 time to be 0. During the weekdays though, the only relevant cycle time components are the move time and the process time. Again, queue time has been eliminated due to modelling, no setups are required, the no batching takes place, and service happens simultaneously for the Batches, so wait-in-batch time is also zero.
The exact move time between the Process 2 and Process 3 is unknown. It is estimated to be [specified time period] at maximum.
The process time depends on [independent variable] within Process 3. Table 3.9 shows estimated

Figure 3.8: Classi ed model.

process times for certain values of [independent variable]. The numbers are based on a model created by another intern at Company X.

Since we did not have access to the full model but only to these data points, we t a function over the data, with the following equation as a result:

$$t_P(i) = 1.13761 + 0.569678 \quad e^{0.051548i} \tag{3.3}$$

With i being the independent variable. More information on the derivation of this equation can be found in Appendix B.2.

[List of assumptions regarding the independent variable].

Now, combining all components, the total cycle time for Process 3 ($_{P3}^{1}(i)$) can be expressed as:

$$_{P3}^{1}(i) = \begin{cases} t_{M(PW2 \quad >PW3)} + t_{PW3}(i) & \text{All weekdays} \\ 0 & \text{Saturday and Sunday} \end{cases}$$

With:

$t_{M(PW2 \quad >PW3)}$: The move time between a PW2 and a PW3, assumed to be [speci ed number].

$t_C(i)$: The process time of a PW3, given by Equation 3.3

Classified

Figure 3.9: Classi ed table.

3.3.5. UCI

Process 4-process is, based on Section 2.6, modelled as follows:

In deriving this model of Process 4, we made the following assumptions and simpli cations:

- Breakdowns of the PW4s were excluded since no su cient data on those events was available.
- The queueing discipline for Process 4 is always Setup Reduction, with Earliest Due Date as second priority. This simpli cation was made based on the fact that this is the usual queueing principle at Process 4 (as described in Section 2.6.
- Setups are always required between the processing of Batches of two di erent orders. There is no data available on how long these setups usually take, so these setups have been modelled to take a constant [speci ed time period] based upon the estimate described in Section 2.6.
- Between the processing of Batches of the same order, a small setup is required in which these Batches are swapped. We observed that this setup commonly takes about [speci ed time period]. During this time period, the employee operating the PW4 can continue working as the setup is carried out by other employees, so the e ective setup time is often shorter. Since this time period is relatively short compared to the other time period within the model and we do not have reliable data on it, we simplify it to be 0.
- The processing speed of the person operating the workstation (the "operator") is constant and consistent throughout their entire shift, i.e. their processing speed does not change over time.
- A PW4 always has the same operator over the full course of a shift.

Classified

Figure 3.10: Classi ed model.

Service process

The service time of the Process 4 is comprised of relatively long queue times and process times, and relatively short move times and setup times. No batching takes place at Process 4, so the wait-to-batch time is zero. With our current model of Process 4, the wait-in-batch time is included in the queue time. The queue time itself is not expressed analytically here. Rather, it arises from running the simulation model based on the presented conceptual model.

The move time between Process 3 and Process 4 is unknown, but it is estimated to be [speci ed time period].

The setup time, as discussed in the Process 4 model description, is modelled to be [speci ed time period] if a setup is required.

Using our assumption that processing speed is constant and consistent, we can model the Process 4 process time P_U of one Batch, similarly to the process time Process 1, as a linear function of the quantity of that Batch:

$$P_P W4(Q) = t_T \quad Q \tag{3.4}$$

With:

Q : The number of Product X in the Batch.

t_T : Takt time of the Process 4.

Now, in order to use this equation to find the process time, we need to derive an expression for the takt time. As described in Section 2.6, the takt times are variable. Since we assume that machine malfunctions do not take place in the model, we reduce the process time components of a Batch to three components:

1. The workstation on which the Batch is being processed.
2. The type of operation to be performed.
3. All other influences, mainly consisting of variability from the operator and of variability due to process errors.

In order to model the takt time, the largest data set available on Process 4 process times [specified time period] was gathered. It quickly became apparent that the data set provided enough information to model component 1 and 2, but component 3 was not represented in the data set. Furthermore, we could not find a statistical distribution to represent the data with sufficient accuracy. As such, we model the takt time as an empirical distribution function, with component 3 modelled as a random variable.
To do so, the data set was divided in separate data sets based on the workstation "$PW4$", and the operation OP (with $OP \in \{OP1; OP2; OP3; OTH; GEN\}$, where "GEN" represents a general distribution in case none of the operations are specified), and for each of those data sets, an EDF $F_{PW4;OP}^{-1}(r)$ was constructed. Here, the variable r represents component 3 ("All other takt time influences") and is pseudo-randomly drawn. Since we assume that an operator's work rate is consistent during their work period and that an PW4 always has the same operator for an entire shift, we can conclude that r is drawn for an entire shift.
Now, implementing our EDF into the process time, we model the service time of Process 4 (${}_{U}^{-1}(r; L; P)$) for the takt time variable in equation 3.4 to obtain:

$$ {}_{P4}^{-1}(r; PW4; OP; Q) = t_{M(PW3->PW4)} + t_Q + t_S + Q \cdot F_{PW4;OP}^{-1}(r) $$

With:

$t_{M(PW3->PW4)}$: The move time between a PW3 and a PW4, assumed to be 10 minutes.
t_Q : The queue time of a Batch at Process 4.
t_S : The setup time (nonzero if Batches of different orders are processed)
Q : The aount of Product X in the Batch.
$F_{PW4;OP}^{-1}(r)$: The takt time at Process 4 during that time, drawn from an EDF.

Chapter 4

Tool design

Central to this chapter is answering sub-question 3:

> "What are the needs and wishes of the problem owner regarding the functioning of a schedule evaluation tool?"

To answer this sub-question, a short interview was conducted to set up a list of requirements. Here, requirements are de ned as follows, based on Aurum & Wohlin [9], p.24:

"A documented representation of a condition or capability as in (1) or (2).

1. *A condition or capability needed by a user to solve a problem or achieve an objective,*
2. *A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, speci cation, or other formally imposed documents."*

The requirements that have been de ned in this chapter serve as a further guide for the development of the tool.
It should be noted that, within professional software development, this step is of key importance and requires a substantial amount of time and thought [9], but due to the time and resource constraints imposed on this bachelor thesis, this step was conducted within a shorter time frame and therefore, the requirements identi cation is less elaborate.
This chapter starts with a description of how requirements should be elicited, followed by multiple sections describing the elicited requirements.

Contents:

4.1. Requirements elicitation methodology

To have a concept of what is expected from a variety of stakeholders regarding this project, a short requirements analysis has been conducted. The most important part of this analysis was to elicit the requirements
According to Aurum & Wohlin [9], requirements elicitation consists of the following steps:

1. Understanding the application domain
2. Identifying the sources of requirements
3. Analyzing the stakeholders
4. Selecting the techniques, approaches and tools to use
5. Eliciting the requirements from stakeholders and other sources

Note that step one has already been performed in chapter 2.
The main sources of requirements and their influence are as follows:

- **The end-user of the schedule evaluation tool**: This stakeholder is arguably the most important source of requirements since they are the only one to actually interact with the schedule evaluation tool. Thus, they are the deciding factor in setting all requirements regarding the interface and most of the functioning of the schedule evaluation tool. The only end-user of the schedule evaluation tool will be the Problem Owner of Company X.
- **The management department of Company X**: This source is not a singular stakeholder, but a group of different stakeholders that can influence the project. They set the constraints on the mission goal of this bachelor thesis based on what they think is valuable for Company X and what is achievable given our current skillset.

The identified stakeholders play a very different role in defining requirements; based on the wishes of management, it has been decided that this project should entail the creation of a schedule evaluation tool, while the end-user should decide how the schedule evaluation tool should function. Management's requirements are already integrated within this bachelor thesis the moment the proposal of this bachelor thesis was approved, which leaves only the requirements of the end-user to be an unknown.
For eliciting the end-user's requirements, Aurum & Wohlin [9] list numerous techniques. It was chosen to elicit requirements by interview and by prototyping. An interview seemed most appropriate because the wishes of only one person needed to be extensively researched (the Problem Owner). Our conducted interview can be found in Appendix A.
Prototyping was chosen to be a supplementary elicitation technique because we assessed during the interview that both we and the Problem Owner did not have a complete picture of the wishes of the Problem Owner, and that their wishes would become clearer once a preliminary version of the schedule evaluation tool could be used. However, the step of testing the prototype has eventually been scrapped from the project due to visiting restrictions as a result of COVID-19. Thus, the requirements of the end-user provided in the next section are only based on the conducted interview. The requirements of the other stakeholders have already been mapped before the start of this bachelor thesis, and thus, no additional research is required.

The following sections describe all requirements that have been elicited from all sources.

4.2. Requirements from the end-user

As has been described in Section 4.1, an interview was conducted for eliciting requirements from the end-user. The method for the interview, the results, the transcripts and the informed consent can all be found in Appendix A.=
From the interview, the following requirements became apparent:

1. The tool should show which PW1s will have shortages, and when.
2. The tool should automatically calculate how many MHE Xs are required for an order.
3. Preferably, the input planning data should be loaded directly from the ERP.
4. The tool should run a simulation with a span of a number of days speci ed on beforehand, and should show which PW1s you would like to see.
5. An Excel-based tool would be su cient
6. Schedule evaluation should preferably not take longer than 10 minutes.
7. During a simulation run, the computer should be available for other tasks, but those do not necessarily have to involve Excel.
8. The tool should function such that the Problem Owner does not have to adapt their way of working.
9. Input parameters for the tool should be adaptable by the user.

4.3. Requirements from the management department of Company X

As has been mentioned in 4.1, the requirements from the management department of Company X have been elicited before the start of this bachelor thesis. In the discussions about the preparation of this thesis, we found the following requirements to be most important:

10. The tool should aid the Problem Owner in reducing the number of MHE X shortage occurrences.
11. If the tool requires an investment, such an investment should be substantiated with an ROI-calculation.

4.4. Self-elicited requirements

Based on experience and common logic, we elicited these additional requirements:

12. The PC of the Problem Owner should be able to run the tool.
13. With normal conduct, no changes should occur to the functioning of the tool.
14. The tool should have Dutch and English directions for use.
15. The communication with and by the tool should be Dutch or English.

Chapter 5

Simulation model

In this chapter, we aim to answer our fourth subquestion:

> "How should the model of Company X's manufacturing process be translated to a DES-model?".

In order to answer this question, we gathered information on DES-simulation, which can be found in Section 5.1. This theory provided a framework that we used to build the simulation model for the schedule evaluation tool. The second section of this chapter therefore provides a description of how the simulation has been developed along with substantiations for those development decisions.

Contents:

5.1. Introductory DES-theory & Contextual information

Discrete Event Simulation (DES) is a simulation type where "only the points in time at which the state of the system changes are represented" [10]. The real world is simulated as a series of instantaneous state-changes, called events. The progression of time is simulated by executing a sequence of events and ignoring the moments in time where nothing happens - between the events - which makes the time progression not continuous but discrete. Which events are carried out is dictated by an event calendar, which is a chronologically ordered list containing all planned events.
Within DES, there are typically two types of events (based on [10]):

B(ooked)-events: These are state changes that are scheduled to occur at a certain point in time, i.e. the events in the event calendar. Within the context of queueing, B-events are usually arrivals or the finishing of an activity.

C(onditional)-events: These are state changes that occur depending on conditions within the model. They can be seen as the reactive state changes. Within the context of queueing, C-events are usually the start of an activity, because starting an activity requires a server to be empty.

There are several different methodologies for DES. For this thesis project, we used the event scheduling method. Within the event scheduling method, the simulation will continuously loop through the following three phases:

A-phase: Find the next B-event and advance the clock to that point.

B-phase: Execute the scheduled B-event.

C-phase: Execute all C-events that logically follow from the B-event.

The advantage of this method is that this method significantly saves computation time. A usual disadvantage of this approach is that it adds complexity [10], but it appeared to be a logical choice for our conceptual model.

In this bachelor thesis, it is attempted to simulate a system that is constantly operative; under normal conditions, there is always WIP in the entire production process, which affects the cycle times of all servers. As such, a simulation model of the production process should have some initial system state to accurately represent the real system from the start. To obtain this initial system state, we introduce a warm-up period, which is a period of time in which the simulation model is run before the actual simulation experiment starts such that a more realistic initial system state is obtained. Section 5.2.4 describes how the warm-up period for this simulation model was derived.

Due to the variability that is being modelled, the output of one simulation run can significantly differ from another. To ensure consistent output results, the simulation is run over multiple replications, and the output is formed from the combined output results of those replications. The derivation of the required number of replications is discussed in Section 5.2.5.

5.2. Simulation development

In this section, we attempt to sketch a picture of how the simulation tool has been developed. We describe what programming language has been used, what the inputs to the simulation are, what the outputs of the simulation are, our derivations for the warm-up period and the required number of replications for accurate simulation results, and lastly, to what extent the requirements specified in Chapter 4 have been fulfilled.

5.2.1. Programming language

For the development of the schedule evaluation tool, it was quickly determined that Visual Basic for Applications (VBA) was the best programming language. While not necessarily optimal for the creation of discrete-event simulations, this programming language had two important advantages:

1. Of all programming languages, we are most familiar with VBA and, since it has many users,

there is enough information on VBA to be found online to consult. As such, choosing VBA substantially accelerates the simulation development process.

2. Since the end-user does not have any experience with programming but uses Microsoft Excel frequently, VBA is an excellent choice because it is built-in Microsoft Excel, allowing for an easy interface in an environment that is familiar to the end-user.

Given these two advantages, we chose VBA as the programming language to develop the schedule evaluation tool.

5.2.2. Input

For the simulation model, we distinguish between three types of input: the candidate production schedule, the input variables and the input parameters. These three input types are discussed in the following two sections. Note that the adaptability of input variables is tied to requirement 9 speci ed in Chapter 4.

Candidate production schedule

On a separate spreadsheet, the user can paste the candidate production schedule. For each order, the user should state the order number, the planned start date, the order quantity, at which PW1 it will be operated and if setups are required at Process 1. It is optional to specify what operation is to be performed at Process 4, the deadline, and the Products per MHE X of an order. If not speci ed, the Process 5 is set to "GEN" (see Section 3.3.5 for clari cation), the deadline is set to the start date [speci ed time period], and the Products per MHE X is set to [speci ed number]. The simulation will start on the earliest day that is noted down in the planned start dates.

Input variables

In order to adapt the simulation to the most up to date settings of the factory, we implemented a user form that allows the user to adapt the simulation settings. This user form initializes at the start of the simulation. The settings that can be adapted are the following:

- Which PW1s and PW4s are operative.
- The starting times and the length in hours of each shift (shift times are sometimes adapted; with the COVID-19 measures by the government in March 2020, the number of shifts got reduced, to decrease the chance of COVID-19 spread among factory workers).
- At which hour of the day the simulation starts.
- Input variable i for Process 3 (see Section 3.3.5).
- The number of available MHE Xs at Company X.
- The number of replications the simulation will run.

The selection menu for the input variables can be seen in Figure 5.1

Classified

Figure 5.1: Classified model.

Input parameters

The end-user can access a sheet with all the input parameters, and can change those input parameters if needed. This sheet contains the following items:

- For each PW1, the takt time and the OEE.
- For each Process 2 setting, the process time.
- For each Process 2 setting, the empirical distribution function of the inter-start time (that is, for each bin of the function, the lower-bound bin and the lower-bound probability)
- For each PW4, the empirical distribution functions for the 5 types of operations as defined in Section 3.3.5 (that is, for each bin of the function, the lower-bound bin and the lower-bound probability)

It should be noted that the empirical distribution functions are quite hard to adapt. However, in case it is necessary to adapt the distributions, we built another tool that creates these empirical distributions automatically from a data set. This tool was programmed because we did not want to create 23 empirical distribution functions by hand.

5.2.3. Output

To generate output, the simulation model stores data on the MHE X usage every [specified time period] in the simulation time. We have chosen a fixed interval for filling data on MHE X usage because it enables easy comparison between the states of two or more simulation runs at the same moment in

simulation time.
To communicate the simulation results to the user, the tool shows a graph at the end of all simulation runs, showing the MHE X usage over time (based on intervals of the specified time period) and the available MHE Xs.
If the simulation was run with two or more replications, the graph shows three lines instead. The first line represents the average number of MHE Xs in use at each interval. The second shows, for each interval, the maximum predicted number of MHE Xs in use. The third line shows, for each interval, the upper bound of the confidence interval for the mean MHE Xs in use at that point in time. Assuming normality of the data, this line is given by:

$$\overline{X(t)} + t_{n-1;\alpha/2} \frac{S(t)}{\sqrt{n}}$$

With:

- $\overline{X(t)}$: the average predicted MHE Xs in use at time t
- $t_{n-1;\alpha/2}$: the value from the Student t-distribution with $df = n - 1$ and the probability of a type I error $\alpha = 0.05$
- $S(t)$: the sample standard deviation of the predicted MHE Xs in use at time t
- n : the number of replications

These three lines provide the end-user with some intuition as to what the result on average might be, and what the worst-case scenario is. An example of the simulation output is presented in Figure 5.2.

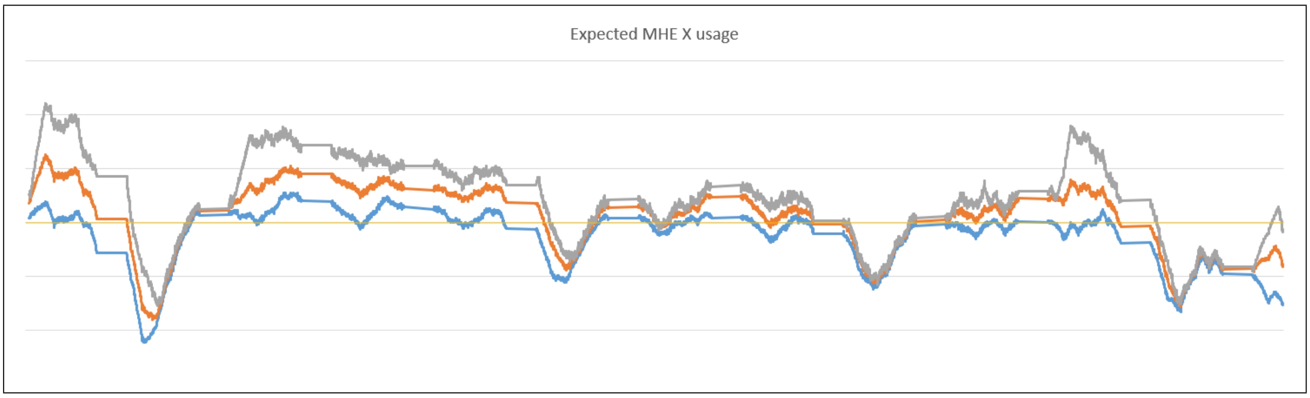


Figure 5.2: Example of the output of the schedule evaluation tool with 10 replications. It can be observed that in multiple instances, the average, upper bound average and maximum MHE X usage exceeds the available MHE Xs. In the real system, this would result in shortages at Process 1.

5.2.4. Warm-up period

Robinson [10] describes several methods to derive a warm-up period for a simulation model. A strong heuristic statistical method to derive the warm-up period is called the Marginal standard error rule (MSER). The aim of this method is to "minimise the width of the confidence interval .. about the mean of the simulation output data following deletion of the initial transient data" [10]. Given an output dataset of $X_1; X_2; ...; X_n$ and a proposed warm-up period d, the MSER-statistic can be calculated as follows:

$$MSER(d) = \frac{1}{(n-d)^2} \sum_{i=d+1}^{n} (X_i - \overline{X}(n;d))^2 \tag{5.1}$$

This statistic should be evaluated for all possible values of d to find the desired warm-up period d that minimizes $MSER(d)$. It should be noted that the statistic becomes unstable for small sample sizes, and as such, values of the MSER for $d \geq n$ should be ignored. Furthermore, to improve the accuracy of the method, it is advised to use averaged data from multiple simulation runs.

In order to derive a warm-up period for the simulation model, we require data from a long simulation run. As such, as input data, a planning horizon should be chosen such that the inputs are consistent over time (e.g. no significant changes in shift times), the planning horizon is not too outdated, and a sufficient amount of data can be gathered. As such, [specified time period] was chosen as the designated planning horizon; [substantiation]. It should be noted that we did not have data on the Products per MHE X and the Process 4 operations of the orders in this planning horizon, which might affect the accuracy of the simulation, but we did not have that data on any series of orders.
The simulation tool was run 20 times over the presented planning horizon. The outputs of the 20 simulation runs were averaged, and the average data was used to calculate all possible values of $MSER(d)$ for $d \in [1; m-5]$, with $m = 4765$ being the number of data filing periods. The result is shown in Figure 5.3.

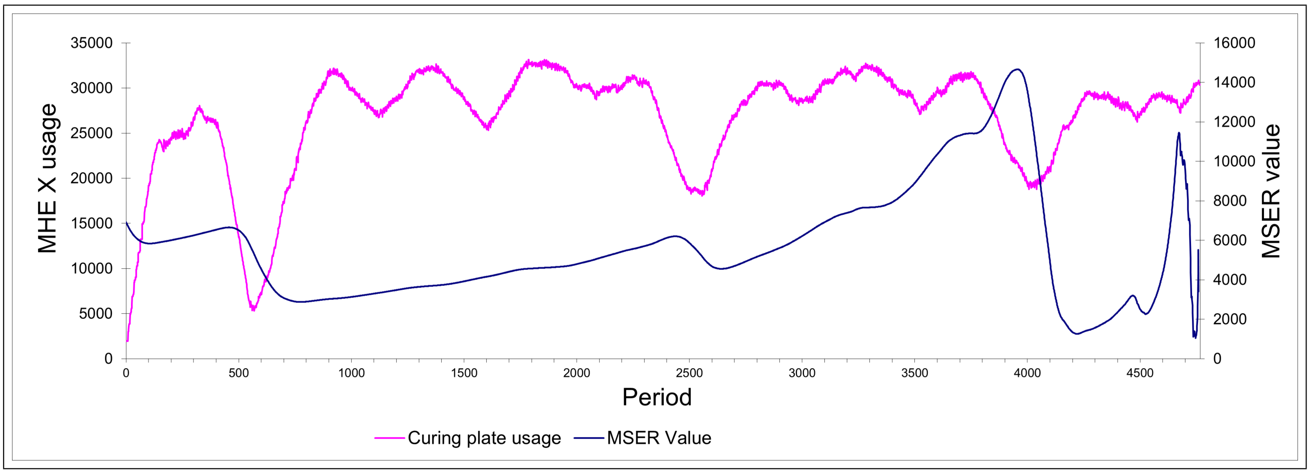


Figure 5.3: Combined graph of average simulation output data over time (purple) and the corresponding MSER values (blue). One can observe local minima for the MSER at period 100, 700, 2600, 4200 and 4500

As can be seen in the figure, there are 5 local minima, with the two lowest minima at the end. This would indicate a warm-up period of at least [specified time period]. However, given our intuitive knowledge of the production system, we think that this warm-up period is unrealistically long and is

probably an error. This error is caused by, as can be seen in the graph of the data in Figure 5.3, the fact that the two minima occur at the last part where the MHE X usage is relatively stable after the last major minimum.
Instead, we chose the period corresponding to the second local minimum observed in the graph of the MSER values to be our warm-up period; safe for the last two minima, this is the lowest minimum that can be observed in the graph, and the data point corresponds to $m = 776 =$ [speci ed time period], which is a more realistic estimation of the warm-up period. To also account for the minimum of the data graph in 5.3 at that period, we set the warm-up period to 9 days.

The warm-up period has been implemented in the simulation model by using the production planning between [speci ed time period] as dummy data.

5.2.5. Number of replications

To derive the least required number of replications, we use the con dence interval width method as proposed by Robinson [10]. For clari cation of this discussion, let $\overline{X}_1, \overline{X}_2, \ldots, \overline{X}_n$ be the average value of the output data of simulation replications $1, 2, \ldots, n$, and $\overline{\overline{X}}$ be the average value of all $\overline{X}_i$, and let $\mu_{\overline{X}}$ be the true mean of the average output data of a simulation run. The con dence interval (CI) around the mean is then given by:

$$CI = \overline{\overline{X}} \pm t_{n-1;\alpha/2}\frac{S}{\sqrt{n}}$$

With

S : the sample standard deviation of $\overline{X}_1, \overline{X}_2, \ldots, \overline{X}_n$

$t_{n-1;\alpha/2}$: the value from the Student t-distribution with $df = n-1$ and the probability of a type I error α

Now, let W be the width of the con dence interval:

$$W = 2 \cdot t_{n-1;\alpha/2}\frac{S}{\sqrt{n}} \tag{5.2}$$

The goal of the method is to minimize the width of the con dence interval such that:

$$E = \frac{W}{\overline{\overline{X}}} < b$$

Where E is a number that we de ne here as the "error", which is the percentage width of the con dence interval, and b is the maximum allowable percentage width of the con dence interval around the mean. For this bachelor thesis, we chose b = 10%.
Assuming that $\overline{\overline{X}}$ remains approximately the same over multiple simulation runs, the only possibility of going below the threshold value b is to decrease W. As can be seen in Equation 5.2, W can be

reduced if n is decreased, since $t_{n-1;\alpha/2}$ decreases if n increases and $\frac{S}{\sqrt{n}}$ decreases if n increases. As such, we need to increase n until the error is reduced below the threshold.
In order to determine at what n the threshold is reached, the simulation was run with 20 replications, and with the results of those replications, the error was calculated for the number of replications $n = 1; 2; \ldots; 20$. The resulting values of the error can be found in Figure 5.4.

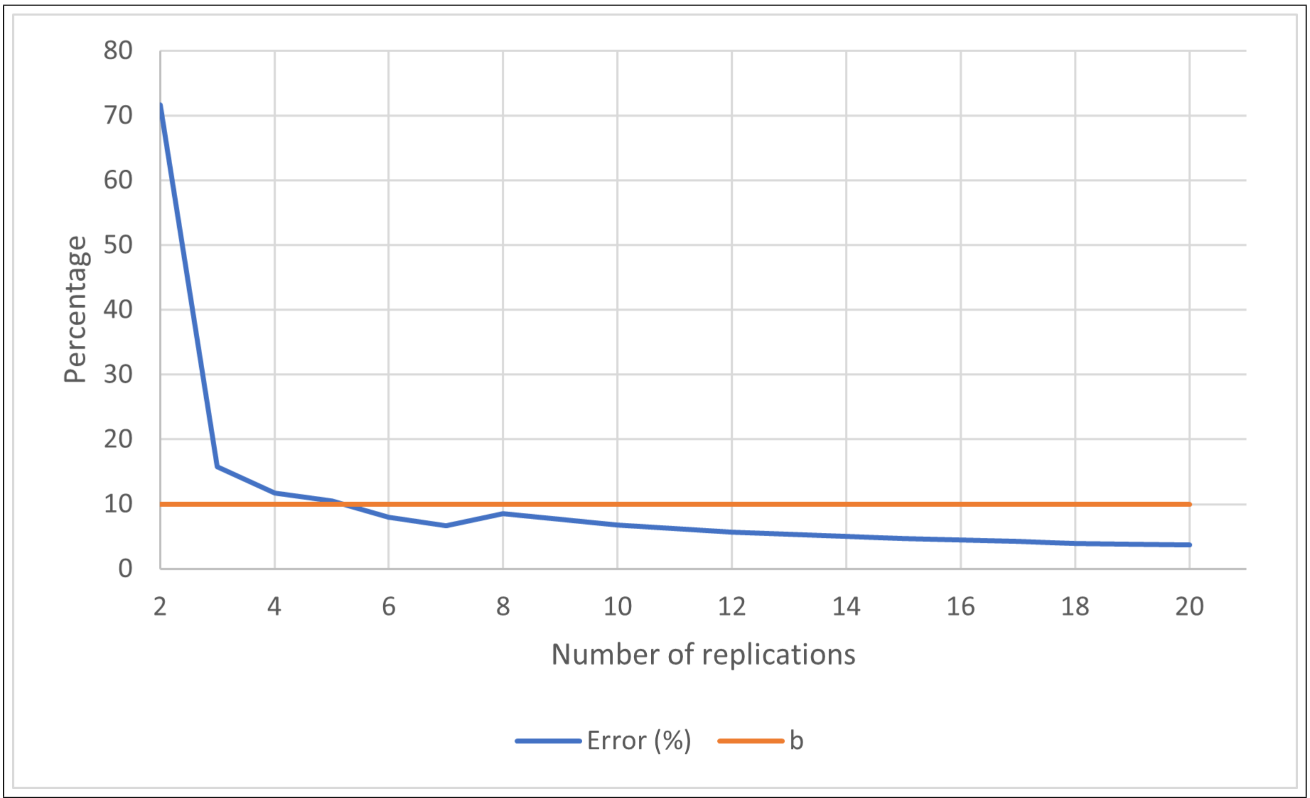


Figure 5.4: Plot of the error (blue line) as a function of the number of replications. The error falls below the threshold after 5 replications.

As can be seen in Figure 5.4, the desired error is already reached at n = 6. However, we observe a local maximum in the error at n = 8, indicating that the error is still not a decreasing function from n = 6 on. For safety, we choose the number of replications to be 10, from which on the error seems to convexly converge to 0.

5.3. Reflection on requirements

Given our current simulation model design, we managed to fulfill most of the requirements stated in Chapter 4. In this section, we discuss the requirements that either have not been fully satisfied or of which it cannot be immediately inferred that they have been satisfied. Requirements that are not discussed in this chapter have been fulfilled.

1. The tool should show which PW1s will have shortages, and when.

As shown in Figure 5.2, the output of the simulation provides insight in at what points in time the MHE X usage is expected to exceed the limit. It should be noted that it does not specify at which PW1s these shortages will occur, but once a shortage occurs in the real system, it affects the entirety of Process 1.

2. The tool should automatically calculate how many MHE Xs are required for an order.

This happens automatically at the start of a simulation run. The user has the option to specify the what the number of Products per MHE X is to increase the accuracy of the simulation.

3. Preferably, the input planning data should be loaded directly from the ERP.

This requirement was dropped because it was quickly identified that implementing this feature would require a considerable amount of time due to our lack of knowledge on how to do this and the lack of access to the ERP. Since it was not a necessary requirement, omitting this feature will affect the quality of the simulation tool but will not compromise the overall goal of the schedule evaluation tool.

6. Schedule evaluation should preferably not take longer than 10 minutes.

At the end of all simulation runs, the schedule evaluation tool shows how many seconds the simulation took. A simulation of 10 runs typically does not exceed 1 minute.

8. The tool should function such that the Problem Owner [(end user)] does not have to adapt their way of working.

The sheet for the input of the candidate productions schedule only requires data that is already available to the end user and is formatted in such a way that most of the data can easily be copied from the ERP and pasted in the tool. As such, no major extra actions have to be undertaken by the end user.

11. If the tool requires an investment, such an investment should be substantiated with an ROI-calculation

No costs were made during the creation of the schedule evaluation tool, safe for some canteen costs.

12. The PC of the Problem Owner should be able to run the tool.

Unfortunately, due to regulations as a result of COVID-19, we haven't been able to test the tool on the PC of the Problem Owner.

15. **The communication with and by the tool should be in Dutch or English**
The front-end of the tool is in Dutch.

Chapter 6

Validation

Before the simulation model is to be adopted by Company X, it should be tested whether the simulation model actually produces valid results. As such, in this chapter, we describe the tests that have been conducted in order to judge whether the created simulation model is su ciently valid. The chapter starts with the selection of a proper Key Performance Indicator (KPI) for testing. The concurrent section explicates the collection process of data on the selected KPI. The third section describes which statistical test was chosen to judge the validity of the model. The nal section summarizes the results of that test, and a conclusion is drawn based on the results.

Contents:

6.1. Key Performance Indicator

In order to compare, we need at least one KPI. The most logical choice for a KPI would be "MHE X availability over time", however, since Company X does not track MHE X availability, this KPI cannot be tested. As Company X does keep track of the progression of an order, we could test the cycle time of an order; Little's Law states that Work-In-Process system, from which the MHE X usage can be derived directly, is proportional to the cycle time of items in that system. To verify if testing using the cycle time is valid, a short test on the correlation between the average MHE X usage and the average order cycle time has been conducted. The conclusion drawn from this test was that there is su cient statistical evidence that there is a correlation between the average MHE X usage and the average order cycle time. Details on this test can be found in Appendix B.3.

6.2. Data collection

In order to make the comparison between the simulation model and the factory, we require data on our chosen KPI on both of the systems. The following two subsections describe how those two data sets were collected.

6.2.1. Simulation data

Testing the similarity between the simulation performance and the factory's performance can only yield valid results if the inputs are the same. Therefore, the simulation should be run for a period of time for which we can be most confident that it accurately mimics the real system. As in Section 5.2.4, we chose the planning horizon to be [specified time period]. See that Section for a thorough justification for choosing this time period. It should again be noted that data on the Product X per MHE X and the Process 4 operations was unavailable, which might affect the accuracy of the simulation.

The input variables regarding the factory itself were set to match the system when it's operative normally (all presses and lines are operative, the factory works with three shifts of 8 hours). The temperature, which has influence on the cooling time, was set to match the average temperature of the given time period (6.7 degrees Centigrade).

Using this setup, the average cycle time over 10 simulation runs was gathered for each individual order. Orders that didn't finish in three or more simulation runs were excluded. In total, data on 525 orders was gathered, resulting in the data set V, with:

V_i : The average cycle time of order *i over 10 simulation runs*

$\overline{X}_V$: The sample mean of all V_i

μ_V : The true mean of the cycle time of each order over 10 simulation runs

6.2.2. Historical data

Company X keeps track of when an order is started at Process 1 and when the last container leaves Process 4 within their ERP system, and as such, the historical cycle time of each order included in the set planning horizon was derived from the data extracted from Company X's ERP system. A notable limitation of the provided data is that Company X only keeps track of the day an order started or finished, and not at which hour, minute or second. A consequence of this limitation is that the cycle time is not guaranteed to be precise; consider the situation where an order starts at Process 1 on day "a" and ends at the UCI on day ""b". From this data follows that the cycle time is $b - a$ days. However, it is unknown at what moment during the day the order was actually started and finished. Consider the following two extreme situations:

Case 1: The order was started on day a at 00:01, and ended on day b at 23:59. The cycle time of the order is approximately $b - a + 1$ days.

Case 2: The order was started on day a at 23:59, and ended on day b at 00:01. The cycle time of the order is approximately $b \quad a \quad 1$ days.

This means that the actual cycle time of that order lies within the range of (b-a-1,b-a+1), which can make a signi cant di erence in assessing the validity of our simulation results, especially since the average order cycle time of the given data is only [speci ed time period] days, which means that each data point on average deviates by [relatively large percentage].
In order to overcome this limitation, we propose to model the cycle data as follows:

$$T_i = t_i + T_{f;i} \quad T_{s;i}$$

With:

T_i :	The actual cycle time in days of an order i
t_i :	The cycle time in days of an order i according to the provided data
$T_{f;i}$:	The moment during the day the order i was nished at the UCI, in days
$T_{s;i}$:	The moment during the day the order i was started at the UCI, in days

Note that $T_{f;i}$ and $T_{s;i}$ are treated here as being stochastic because we do not know what their actual values are. As a consequence, T_i is stochastic as well. Now, if we assume that $T_{f;i}$ and $T_{s;i}$ are both U(0,1)-distributed (i.e. there is an equal probability for each moment of the day that an order starts at the presses or nishes at the UCI), then E[$T_{f;i}$] =E[$T_{s;i}$] = 0:5. As a consequence, E[T_i] =E[$t_i + T_{f;i} \quad T_{s;i}$] =E[$t_i$]+E[$T_{f;i}$] E[$T_{s;i}$] = t_i.
With this model, we can say that the actual cycle time of an order is, on average, equal to the cycle time featured in the data. We use this as a justi cation to use the cycle time of each order featured in the data to represent the actual cycle time, but it should be noted that this is by no means valid. Using this model, we constructed the data set R with:

R_i :	The cycle time of an order $i$according to the provided data
$\overline{X}_R$:	The sample mean of all R_i
$_R$:	The true mean of the cycle time of orders from the historical data

6.3. Test

Conventional statistical tests that are used for the comparison of two populations are the t-test for comparison of two means, and the paired t-test for comparison of two means. These tests can only be used if it can reasonably be assumed that there is normality of the data. We assume normality of the data, and our justi cation for that can be found in Appendix B.4.
If there is correlation between the two populations, the paired t-test is more appropriate. Since the historical data and the simulation data have similar inputs (the simulation data has been generated using the same production planning as the historical data), correlation between the historical data and the simulation data can be assumed. To verify correlation between the historical and simulation data, a statistical test has been conducted. The conclusion of this test was that there is statistical

evidence that there is a correlation between the simulation data and the historical data. Details on this test can be found in Appendix B.5.
We conducted a two-sided paired t-test with $\alpha = 0.05$ (the probability of a type I error), testing H0: $\mu_R = \mu_V$ against H1: $\mu_R \neq \mu_V$. Then, the test statistic T of the paired t-test is given by:

$$T = \frac{\overline{x}_d - 0}{\frac{S_d}{\sqrt{n}}}$$

With $\overline{x}_d$ being the sample mean of the differences between the simulation cycle time and the historical data of orders, S_d being the corresponding sample standard deviation, and n being the sample size. T has a t_{n-1}-distribution under H_0. The results of the paired t-test have been reported in the next section

6.4. Test result & conclusion

With $\alpha = 0.05$ and degrees of freedom $df = n - 1 = 524$, the critical value c for the paired t-test is $c \approx 1.964$. As for our test statistic T, we found $T \approx 10.461$. Since $c < T$, we conclude that there is significant statistic evidence to reject H0. This means that, with a confidence level of 95%, that there is a significant deviation between the simulation results and the historical data, and that the simulation most likely does not represent the real world accurately.
For the sample mean of the differences $\overline{x}_d$, we found $\overline{x}_d = \overline{x}_R - \overline{x}_V = 0.732$, indicating that the predictions by the simulation for the order cycle times are optimistic.
As for the sample standard deviation x_d, we found $x_d = 1.535$. Since this is a significant percentage of the average cycle time of the historical data [substantiation], this number indicates that, between the differences of the historical cycle time and the simulation cycle time of all the orders, the differences feature relatively large variance. This might be another indicator that the accuracy of the simulation should be improved.
It should be noted that these conclusions are not entirely complete. As discussed in Section 6.2.2, the historical data is expected to be inaccurate; for an order with cycle time x, the true cycle time of that order lies within the interval (x-1, x+1). If the true historical cycle time of each order is on average 0.793 days shorter than $\overline{x}_R$, then $\overline{x}_d = (\overline{x}_R - 0.793) - \overline{x}_V = 0$, which results in the test statistic $T = 0$, for which H_0 would be accepted.
By performing a goal seek, we tried to find the allowable range for the difference between the actual historical cycle times and the cycle times in our data set R such that H_0 would not be rejected. Assuming that S_d remains constant, we found that, given $\overline{x}_d = (\overline{x}_R - E) - \overline{x}_V$, H_1 is rejected if $E \in (0.556, 0.812)$. This means that, if the historical cycle times in our data set R are structurally overestimated by 0.556 to 0.812 days, our rejection of H_0 might not be correct.
Further limitations to the data include that we do not know the Product X per MHE X and the Process 4 operations used for each order of the historical data. Missing this vital input for the simulation could also have resulted in a compromise of the accuracy of the simulation.
In conclusion: there is statistical evidence that the simulation does not represent the real world accurately. However, the quality of this statistical evidence is disputed, and if the actual real life cycle times are (significantly) shorter than the cycle times in our historical data set, this statistical evidence

is sure to be incorrect.

Chapter 7

Conclusion

This bachelor thesis was initiated by to solve the following problem Company X encountered in their production:

> "The number of MHEs required for running production is not properly considered in the planning of the production process."

In order to remedy this problem, it was attempted to develop a schedule evaluation tool for Company X that could predict MHE availability over time given a candidate production schedule, using discrete-event simulation. For the development of such a tool, we constructed the following research question:

> ***"How should a DES-based tool that predicts MHE X requirements resulting from a candidate production schedule be developed?"***

To answer this research question, it was split into four sub-questions:

SQ 1: What processes are involved in the production of Product Xs at Company X?

SQ 2: How should Company X's manufacturing process for the creation of Product Xs be modelled?

SQ 3: What are the needs and wishes of the Problem Owner regarding the functioning of a schedule evaluation tool?

SQ 4: How should the model of Company X's manufacturing process be translated to a DES-model?

This chapter serves as a re ection on the process of answering these questions.
In the rst section of this chapter, we discuss whether the objectives of this bachelor thesis have been met. In the section afterwards, we provide recommendations for the improvement of the schedule evaluation tool.

7.1. Discussion

In this section, we rst discuss whether the research question and the sub-questions have been answered satisfactorily. Afterwards, we discuss whether the stated core problem hss been solved satisfactorily.

7.1.1. Research questions

On rst sight, this thesis provides an exhaustive answer to our research question; chapter 2 thoroughly answers sub-question 1, sub-question 2 is answered in Chapter 3, Chapter 4 describes our answer to sub-question 3, and in Chapter 5, our last sub-question is answered. The combination to these answers provide the reader with a strong framework that can be used to develop a tool similar to the one developed in this thesis (a DES-based tool that predicts MHE X requirements resulting from a candidate production schedule).
While we think that our answers to sub-question 1 and sub-question 4 were complete, we are unsure whether that is the case for our answers to sub-question 2 and 3. As such, they are discussed separately in the following two sections.

Sub-question 2: How should Company X's manufacturing process for the creation of Product Xs be modelled?

As concluded in Section 6.4, there is statistical evidence that the simulation model does not accurately represent the production process of Company X yet, and that the simulation is on average more optimistic about the cycle times of orders. From this, we infer that the conceptual model of the production process was most likely inaccurate, indicating that our modelling method was incorrect.
We recognise that this could be the case; in our attempt to model the production process, we made some major assumptions and simpli cations that could a ect the results of the simulation model to such an extent that it is not anymore an accurate representation of reality. The most potentially invalid simpli cations are the following:

1. The PW1s are modelled as servers with a constant takt time and no breakdowns.
2. It is expected that our current model of Process 2 yields lower inter-start times on average.
3. The Process 2 and Process 3 are modelled such that no queueing takes place.

The combination of these three simpli cations could de nitely have caused the simulation results to be more optimistic than reality, although it should be noted that these simpli cations were a necessary evil for the model development as we lacked the data to model the underlying phenomena more accurately.
However, as concluded in Section 6.4, the correctness of the statistical evidence against the validity of the model is debatable; due to a lack of precise data, the validity of the statistical test results could have been compromised. As such, we cannot grant a nal verdict as to whether sub-question 2 was answered correctly.

Sub-question 3: What are the needs and wishes of the Problem Owner regarding the functioning of a schedule evaluation tool?

While Chapter 4 provides a list of requirements for the simulation tool, we suspect that this list might be incomplete or inaccurate; as discussed in Section 4.2 and Appendix A, the quality of the interview is debatable, and as such, we might have missed some requirements. However, the interview was structured as such the goal of the tool was communicated clearly, and the interviewee, the Problem

Owner, had multiple opportunities to communicate their most important wishes. From this, we infer that the most important requirements have been gathered.
As proposed in Section 4.1, we could have obtained a more accurate list of requirements by conducting a prototype test with the Problem Owner. However, by the time the schedule evaluation tool was finished, we were unable to conduct this test in person due to COVID-19. As such, we currently don't know whether the created schedule evaluation tool meets all expectations.

7.1.2. Solving the core problem

The goal of this bachelor thesis was to solve the MHE shortages at Company X. To reduce the workload for this thesis such that it fits within the projected time frame of a bachelor thesis (420 hours), the scope of this thesis was reduced to only MHE Xs. The intention was to develop a schedule evaluation tool that could be used to predict when MHE X shortages would take place given a candidate production schedule so that the candidate production schedule can be adapted such that no shortages take place.
The tool that was developed for this bachelor thesis is able to generate predictions on whether shortages occur given a candidate production schedule. In that regard, we managed to reach the specified goal. However, as discussed in the previous sections, it is unsure whether the output of the schedule evaluation tool is representative for the real world. It should be noted that, since the simulation results seem to structurally underestimate the MHE X usage, the tool could be used to predict peaks in MHE X usage instead.
In conclusion: we for now counsel against taking the results of a simulation run as full truth and basing policy on the results of the schedule evaluation tool. Instead, we advise to use the tool as a second opinion on potential overscheduling until the results of the simulation tool are verified to be consistently valid. In the next section, we describes what steps should be undertaken to ensure that the core problem can be solved once and for all.

7.2. Recommendations

For further development and improvement tool, we recommend that several activities, or steps, should be undertaken. These steps are schematically represented in Figure 7.1. The remainder of this section discusses each step in more detail. As a first step, we recommend that the outputs of the simulation should be validated with more precise data so it can be determined whether the schedule evaluation tool is an accurate representation of the real production system. To do so, more precise data on the cycle time of orders needs to be gathered, with which the testing procedure described in Chapter 6 can be repeated to conclude whether the schedule evaluation tool can be adopted.
If this first step is taken and it turns out that the results are not accurate, we recommend to improve on the current conceptual model. Specifically, the stochasticity of the Process 1 takt times and the time between PW2 starts should be investigated more thoroughly. The former requires data to be collected on the actual press times and yields and the occurrence of breakdowns, and the latter requires Company X to investigate the distribution of the inter-oven times for all ovens that started without being full. We think that, with the insights gained from these data analyses implemented,

the simulation will de nitely produce outputs that represent reality.

To fully solve the core problem, we recommend that the schedule evaluation tool is extended to include all MHE types in the model. Note that adding this feature also likely bene ts the accuracy of the simulation output because all material streams are then featured in the model, which eliminates the necessity of the simpli cation that no queueing takes place at Process 2 or Process 3. To implement this feature, a substantial amount of data needs to be gathered on the material streams of other MHE types, and the conceptual model and the simulation model need to be adapted. This project requires a signi cant amount of time, and if it to be undertaken, we suggest that it it is to be conducted as a Masters thesis.

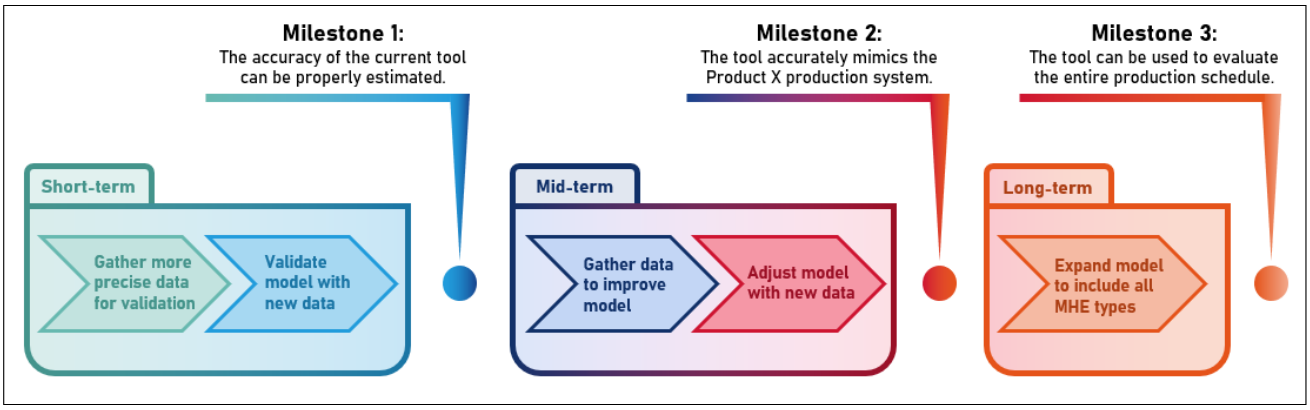


Figure 7.1: Road map showing the recommended steps to be undertaken for further development of the schedule evaluation tool.

Bibliography

[1] V. Tardif and M. L. Spearman, \Diagnostic scheduling in nite-capacity production environments," *Computers & Industrial Engineering*, vol. 32, no. 4, pp. 867 - 878, 1997.

[2] S. L. D. Haro, S. B. Gershwin, and D. B. Rosen eld, \Schedule evaluation in unstable manufacturing environments," *International Journal of Production Economics*, vol. 121, no. 1, pp. 183 - 194, 2009.

[3] B. Kadar, A. Pfei er, and L. Monostori, \Discrete event simulation for supporting production planning and scheduling decisions in digital factories," in *37th CIRP International Seminar on Manufacturing Systems: Digital Enterprises*, 2009, pp. 444 - 448.

[4] P. Appelqvist and J. M. Lehtonen, \Combining optimisation and simulation for steel production scheduling," *Journal of Manufacturing Technology Management*, vol. 16, no. 2, pp. 197 - 210, 2005.

[5] A. Graunke, G. Burnett, C. Hu, and G. Wirth, \Decision support model to evaluate complex overhead crane schedules," in *Journal of Manufacturing Technology Management*, 2014.

[6] G. E. Vieira, M. Kuck, E. Frazzon, and M. Feitag, \Evaluating the robustness of production schedules using discreteevent simulation," *IFACPapersOnLine*, vol. 50, no. 1, pp. 7953 - 7958, 2017.

[7] W. J. Hopp and M. L. Spearman, *Factory Physics*, 3rd ed. Long Grove, IL: Waveland Press, Inc, 2008.

[8] P. Bratley, B. L. Fox, and L. E. Schrage, *A guide to simulation*. New York: Springer, 1987.

[9] A. Aurum and C. Wohlin, *Engineering and managing software requirements*. Berlin: Springer, 2011.

[10] S. Robinson, *Simulation - The practice of model development and use*, 2nd ed. New Jersey: Wiley, 2014.

Appendix A

Interview

Note to the reader: the transcript of the interview was removed for anonymity purposes.

This appendix contains a short description of the short interview study conducted with the Problem Owner of Company X to discover requirements to be set for the development of the schedule evaluation tool.
The initial goal of the simulation is described in appendix A.1. Appendix A.2 contains a description of the interview setup. Appendix A.3 then shows what requirements have been set up based upon the interview.

A.1. Interview goal

The goal of this interview is to investigate the needs and wishes of the Problem Owner regarding the functioning of the schedule evaluation tool. This information will be used for mapping the preliminary requirements for the development of the schedule evaluation tool. The most important information to be gathered is how the Problem Owner wishes to interact with the schedule evaluation tool. This includes: how they want to provide the tool with input, what kind of output they wish to see and what happens in the process of generating output. Furthermore, requirements that influence the type of software should also be mapped.

A.2. Method

This section aims to describe how the interview will be conducted. First, the setup will be discussed, which includes all important characteristics of the interview. Afterwards, the interview structure will be reported. The section ends with a list of the questions that will require answering.

A.2.1. Design

To achieve the interview goal, the Problem Owner of Company X will be the subject of the interview, as he is assumed to be a major expert regarding his own wishes. To discover what these wishes are, a semi-structured interview seemed most appropriate; while there are several specific topics that need to be discussed, it could be possible that there are some unanticipated wishes that the Problem Owner could express that can be explored through probing. The interview itself will be held in Dutch, because it is the mother tongue of both the interviewer and the interviewee, and it is the customary language at Company X. The interview will be conducted by the author of this bachelor thesis. The interview will be conducted face-to-face because it was the most convenient option available; it just so happens that my desk is situated next to the desk of the interviewee. Data will be collected through note-taking and audio recordings; it is anticipated that some questions are rather technical and require dialogue for clarification, and during these clarifications, it is hard to take accurate notes. The data will then be described in clean-verbatim; stutters, filler speech, non-speech sounds and interjections by the interviewer (such as encouraging words) have no significant impact on the conclusions of this interview and are thus left out of the transcript for clarity.

A.2.2. Structure

Prior to the interview, the goals of this thesis project should be discussed to set the correct perspective; the Problem Owner has multiple daily tasks and is not closely involved with this project.
For the interview itself, the questions as described in Appendix A.2.3 will be asked. However, the interview will start by inquiring how the Problem Owner envisions the tool should work. This question is asked to ensure that the interviewer does not significantly influence the perspective of the Problem Owner on the schedule evaluation tool. The idea is that a broad picture of the functioning is described. After this broad investigation, the questions described in the next section will be asked to provide the leftover required information.

A.2.3. Questions

Below, the questions that were planned to be answered with this interview are listed. The questions will not necessarily be asked in this specific order. For each question, a short description has been provided as to why the question is relevant.

- *Hoe zou jij het liefst willen dat de input-gegevens in de tool worden gezet?*
  If a specific input format is required, it could influence the software choice. Furthermore, the desired input format determines how the tool should be designed.
- *Hoe zou jij willen dat de output wordt gevisualiseerd?*
  Specific output format wishes might require features that are possible in one but not another software program. Furthermore, this question is asked to determine how the tool should be designed.
- *Heb jij voorkeuren omtrent in welk programma de tool wordt gerund?*
  This question gives the interviewee the opportunity to express preferences with regards to the software choice.
- *Hoe lang mag het totale simulatieproces duren?*
  Required simulation duration could influence the type of software used as well as the depth of the simulation.

- *Hoeveel acties zou jij willen doen voordat de simulatie start?*
  The desired amount of actions in uences the simulation layout.
- *Op het moment dat de simulatie draait op jouw computer, zou het erg zijn als je Excel niet zou kunnen gebruiken?*
  The Excel-sheet in which VBA is running cannot be used during the run. Thus, if the Problem Owner would like to use Excel during the runtime, VBA becomes less desirable.
- *Hoeveel zou het resultaat van de simulatie mogen afwijken van de realiteit?*
  The required level of accuracy in uences the depth of the model.
- *In hoeverre zou jij je werkzaamheden willen aanpassen om de tool goed te laten werken?*
  The production planning horizon is currently [speci ed time period]. A longer planning horizon might be desired for simulation accuracy.
- *Welke onderdelen van de simulatie zouden aangepast moeten kunnen worden?*
  The answer to this question helps determining how the input of the tool should be regulated.

A.3. Results

The raw data obtained from the interview [has been deleted from this thesis version]. Within the transcript, some answers have been labeled with a number. Based upon those labeled statements, the following conclusions have been drawn (with conclusion number corresponding to the label):

1. The tool should show which presses will have shortages, and when.
2. The tool should automatically calculate how many MHE Xs are required for an order.
3. Preferably, the planning data should be loaded directly from their ERP.
4. A simulation run should span a period of [speci ed time period].
5. A simulation run should show which PW1s you would like to see.
6. An Excel-based tool would be su cient. The author does not want to draw any conclusions regarding other software programs because they are unsure whether the the Problem Owner was su ciently informed.
7. The total schedule evaluation should preferably not take longer than 10 minutes.
8. During a simulation run, the computer should be available for other tasks, but those do not necessarily have to involve Excel.
9. The Problem Owner has expressed no speci c wishes with regards to accuracy. The author draws the conclusion that high levels of accuracy are not required.
10. The Problem Owner does not want to adapt their way of working to bene t the tool.
11. The Problem Owner did not express any speci c wishes regarding the format of the output. A graph would be su cient, as long as it shows when shortages will take place and at which presses.
12. It would be useful if parameters can be changed within the tool. Speci cally: time between PW2 starts, Process 2 service times, Process 3 times and the number of machines. If it is

not difficult, the Problem Owner would like to be able to do change these parameters by themselves.

Based upon these results, requirements have been set up for the schedule evaluation tool in Chapter 4.

A.4. Validity

When analyzing the transcript, it can be argued that the validity of this interview is questionable; some of the questions asked were suggestive and might have influenced the answers of the Problem Owner. Moreover, we are unsure whether our questions asked matched what they actually wanted to know. It is therefore recommended that these results should not be used in research outside of this thesis project. It should be noted, however, that these results can be used to set up requirements for this thesis project; the main consequence of these results being wrong is that the author creates a schedule evaluation tool that does not match with the expectations of the Problem Owner. This would cause a significant delay in the thesis project, but we accept the risk.

Appendix B

Data analyses

This appendix features all calculations or tests that have been performed to provide with substantiation to some assumptions made throughout this Bachelor Thesis.

B.1. Estimating the probability of queueing at Process 3

[Referred to from Section 3.3.4]

In order to estimate the probability of queueing at Process 3, we used data on Process 2 finishes, along with Table B.1 (originally represented in Table 3.9), to estimate what percentage of the times more than three Process 2 batches are present at Process 3. As has been discussed in Section 2.5, there are only three PW3s that each can fit up to one Process 2 batch, so whenever there are more than three Process 2 batches present at Process 3, queueing should occur.

In order to provide with an estimate, we used a data set containing all Process 2 finishes between [specified time period]. With this data set, we calculated the time difference between a PW2 finish (t_i) and the PW2 finish of the previous three Process 2 batches (t_{i-3}) for all Process 2 batches that time frame. $t_i - t_{i-3}$ represents the maximum amount of time Process 2 batch $i - 3$ can sojourn in Process 3 before queueing takes place (i.e. when Process 2 batch i arrives at Process 3). Afterwards, we used the values in Table B.1 to determine what proportion of the time $t_i - t_{i-3}$ exceeded the sojourn time of one Process 2 batch.

With this estimation paradigm, we made the following implicit assumptions:

- One Process 2 batch always fills one PW3.
- Move time and Process time is the same for every Process 2 batch.
- Whenever an Process 2 batch is finished, it is transported immediately to Process 4.
- Every container has to be processed the same amount of time in Process 3.
- The chosen data set is representative for the future.

The data set contained a total of 837 data points. There were 55 cases in which $t_i - t_{i-3}$ was longer than [unrealistic value]. These were excluded because these are [substantiation for unrealism], and do not represent a fully operative production system.

It was found that less than 3% of the time, $t_i \quad ti \quad 3$ was lower than ordinal value $i = 7$. [Values higher than ordinal value 7 are fairly uncommon]. Thus, with the stated assumptions simpli cations, the probability of queueing is estimated to be below 3%.

i	P(Queue)
0	0.25%
1	0.51%
2	0.51%
3	0.76%
4	1.02%
5	1.53%
6	2.04%
7	2.93%

Table B.1: Estimated queue probability at Process 3 based upon the variable i. i is presented here as an ordinal variable; the actual variables of i have been removed.

B.2. Deriving a function for the Process 2 service time

[Referred to from Section 3.3.4]

In order to implement a decent model for the Process 3 service times in our conceptual model, we required a continuous function of the Process 3 service times. Unfortunately, the original model used by Company X to derive the Process 3 service time data itself was unavailable. As such, we attempted to mimic the original model underlying the data by tting a function using a least-squares t.
We plotted the data to see what type of function would t the data the best, and we quickly concluded that a exponential function was most appropriate. Using a least-squares t function from Mathematica, we obtained the following result:

$$t_{Co}(T) = 1{:}13761 + 0{:}569678 \quad e^{0{:}0515487T} \tag{B.1}$$

As depicted in Figure B.1, this function is a near-perfect t.
We do note that this methodology has a limitation: [limitation with conclusion: it is unlikely that this model accurately represents data outside of $i\ 2\ 0; 7$].

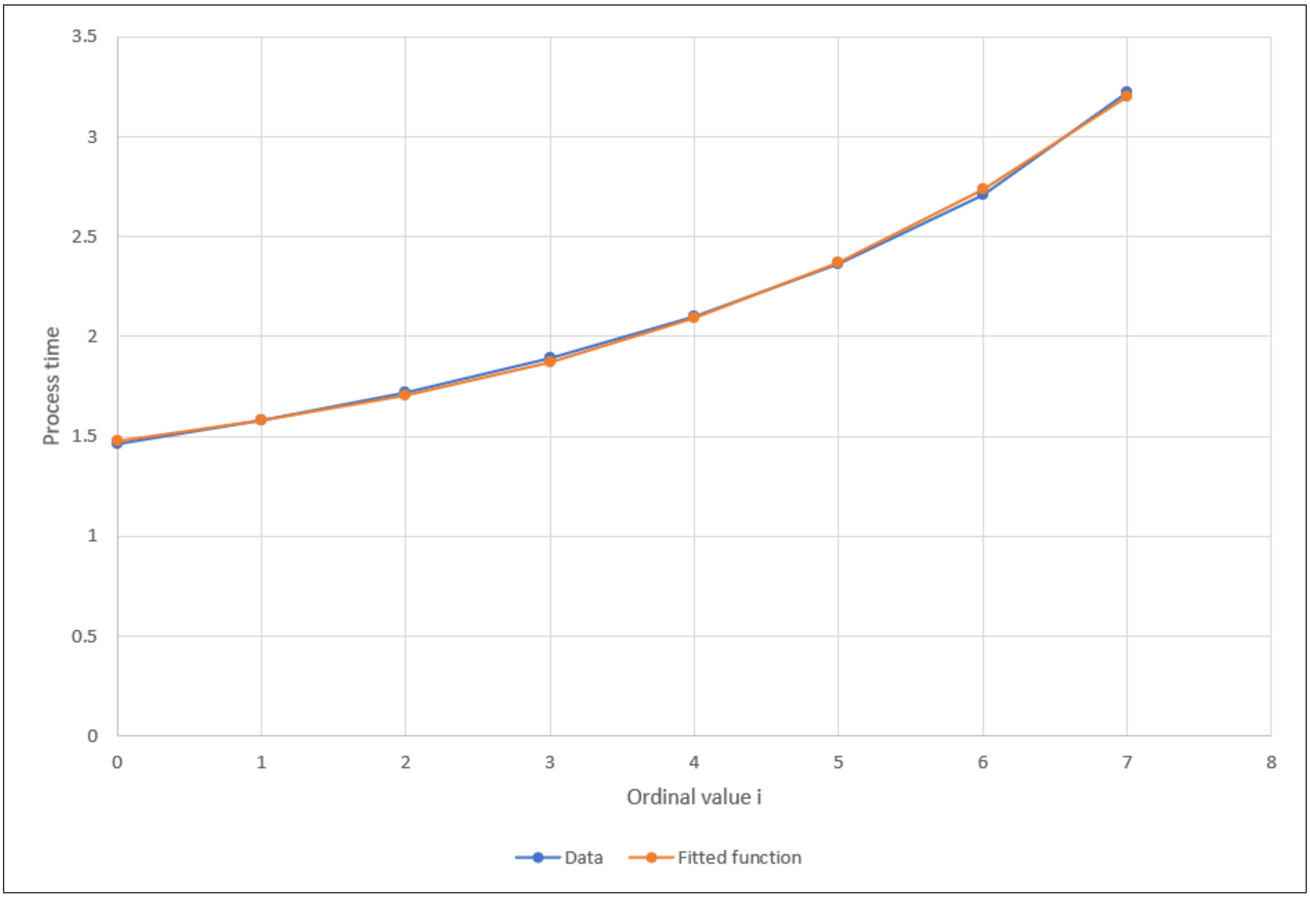


Figure B.1: Plot of the expected Process 3 times against the fitted function.

B.3. Test for correlation between MHE X and cycle time

[Referred to from Section 6.1]

In order to test whether order cycle time is a valid testing KPI, we investigated whether there was sufficient correlation between MHE X usage and order cycle time. To do so, we performed 30 simulation runs, of which we collected for each run the average MHE X usage and the average order cycle time. A scatter plot showing the potential correlation between the data of the simulation runs can be found in Figure B.2.

Subsequently, we performed a t-test on the significance of the (Pearson's) correlation coefficient (ρ). We tested $H_0 : \rho = 0$ against $H_1 : \rho \neq 0$, with the probability of a Type I error $\alpha = 0.05$. The test statistic corresponding to the t-test on the significance of the correlation coefficient is:

$$T = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}}$$

Where $n = 30$ represents the sample size.

Based on the data of the 30 simulation runs, we found $r \approx 0.94$, with a corresponding $T \approx 14.73$. Under H_0, T has t_{n-2}-distribution. The critical value of the t-distribution for $\alpha = 0.05$, $df = n - 2 = 28$ and a two-sided test is $p \approx 2.05$. Since the value of T is larger than the critical value, we reject H_0.

As such, we conclude that there is statistical evidence that the correlation between average MHE X usage and average order cycle time is signi cant, and that, as a result, we can use the cycle time as our testing KPI.

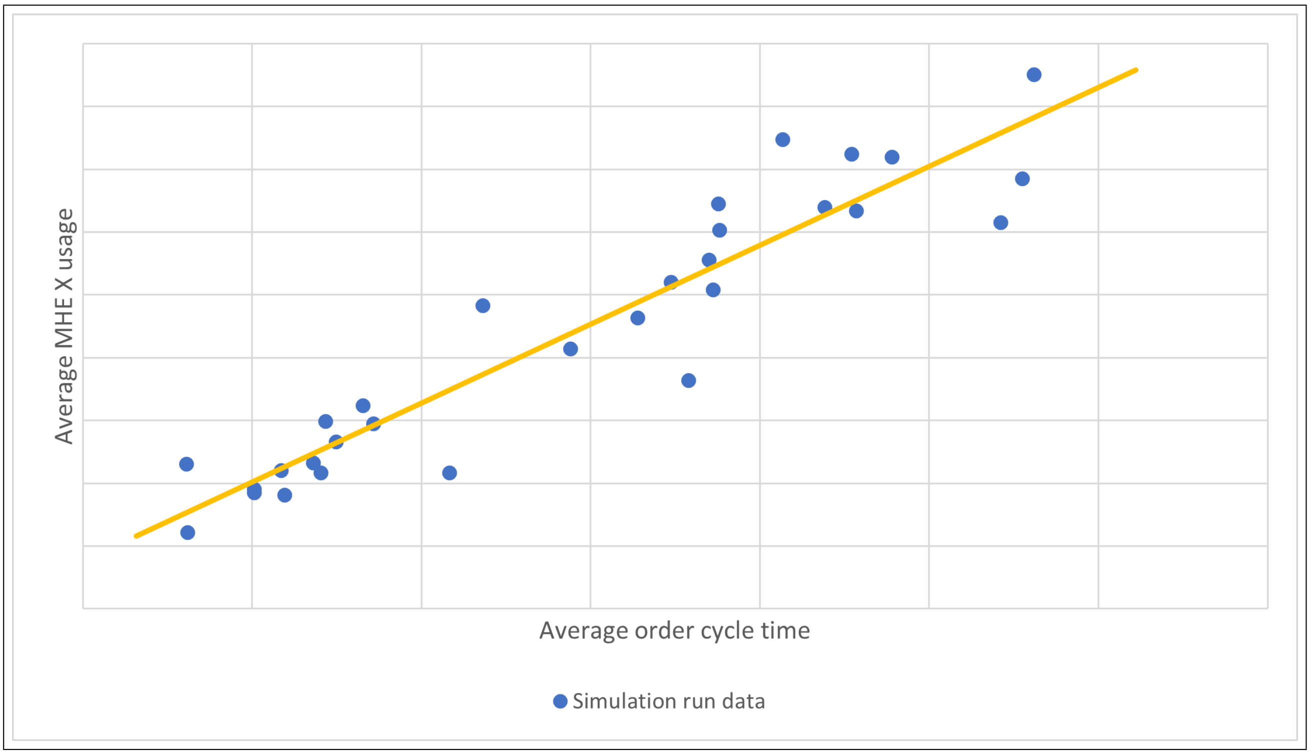


Figure B.2: Scatter plot of the average order cycle time and average curing plate usage of 30 simulation runs. The yellow line is a linear trend line drawn through the data.

B.4. Normality of the validation data

[Referred to from Section 6.3]

In order to use a paired t-test for validation, the data points in the sample used for our validation test is required to be normally distributed. Here, the data points we refer to are the di erences between the simulation cycle time and the real life cycle time for each order.

As a simple initial test, we inspected the values of the skewness and kurtosis of our sample. Normal distribution typically have skewness values around 0 and kurtosis values around 3. Based upon the data, we found a skewness of 0:598 and a kurtosis of 4:088. Since these values do not di er signi cantly from the standard values for a normal distributions, we see no reason to reject the assumption of normality of the data.

To ensure that the data approximates a normal distribution, we performed a short visual test by plotting a histogram of the validation data against an example normal distribution. The result can be found in Figure B.3. While not a perfect t, the distribution of the data seems to approximate a normal distribution.

Based upon the two super cial tests, we have found no convincing motive to conclude that the data is not normal. As such, we appeal to the Central Limit Theorem; the Central Limit Theorem states that, given a large enough sample of random variables (usually $n > 25$), the distribution of those random variables tends to approximate a normal distribution. Since our initial tests indicated an approximate normal distribution, we conclude that the distribution underlying the data resembles a normal distribution enough such that we can assume normality of the data under the Central Limit Theorem. As such, we can use a t-test to test whether there is a significant difference between the historical data and the simulation data.

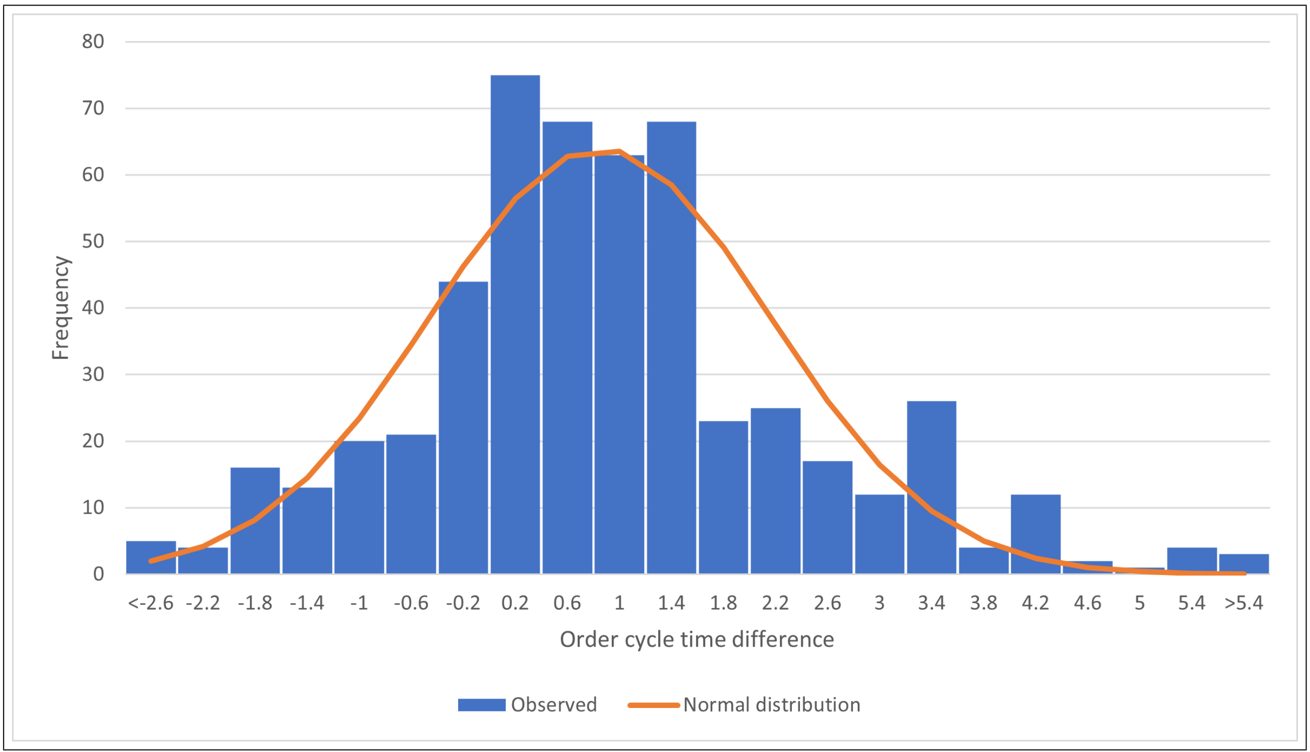


Figure B.3: Combined plot of a histogram of the validation data and an example normal distribution function.

B.5. Test for correlation between historical order cycle times and simulation order cycle times

[Referred to from Section 6.3]

In order to use a paired sample t-test, there needs to be sufficient evidence that the two samples are correlated. As such, we performed a t-test on the significance of the (Pearson's) correlation coefficient with the historical cycle time data and the simulation cycle time data. We tested $H_0 : \rho = 0$ against $H_1 : \rho \neq 0$, where ρ represents the Pearson's correlation coefficient, and with the probability of a Type I error $\alpha = 0.05$. The test statistic corresponding to the t-test on the significance of the correlation coefficient is:

$$T = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}}$$

Where $n = 526$ represents the sample size.
Based on the validation data sample, we found $r \approx 0.38$ (the sample correlation coefficient), with a corresponding $T \approx 9.36$. Under H_0, T has t_{n-2}-distribution. The critical value of the t-distribution for $\alpha = 0.05$, $df = n - 2 = 524$ and a two-sided test is $p \approx 1.97$. Since the value of T is larger than the critical value, we reject H_0. As such, we conclude that there is statistically significant correlation between the historical cycle time of each order and the simulation cycle time of each order, and that we can use the paired t-test for our validation test.
For a visual representation of the correlation, we refer to Figure B.4.

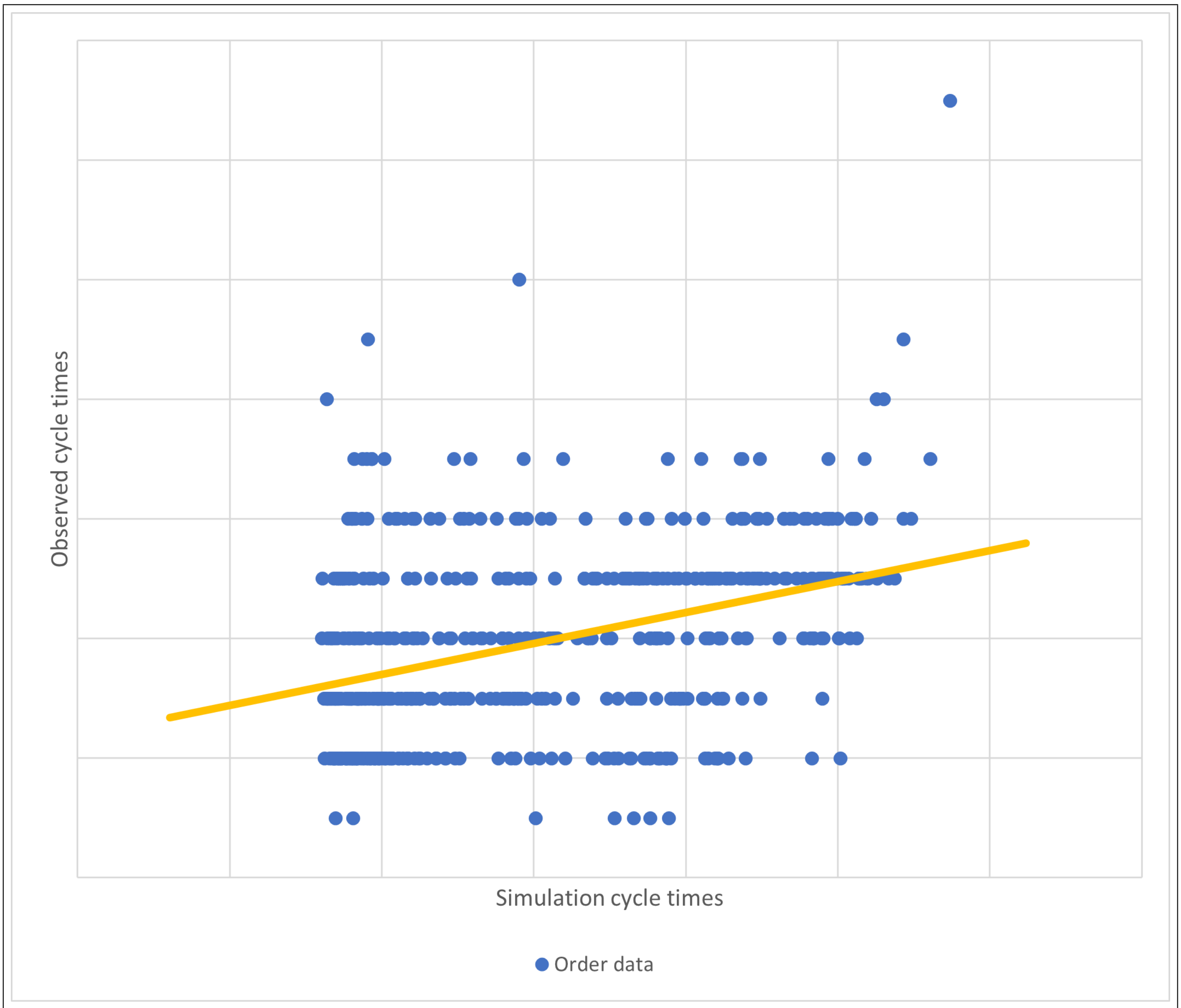


Figure B.4: Scatter plot of the historical order cycle time and the simulation order cycle time over a total of 526 orders. The points appear in horizontal lines because the simulation data is precise to the second while the historical data is only precise to the day. The yellow line represents a linear trend line drawn through the data. It should be noted that this plot approximately resembles a map of the United States