Sound management at Comfoor

REDUCING THROUGHPUT TIME BY SIMULATION BASED ON MATHEMATICAL CAPACITY MODELLING

Lisanne van Zadelhoff
UNIVERSITEIT TWENTE IN SERVICE OF COMFOOR B.V. | ENSCHEDE

Comfoor B.V.
Vlijtstraat 60-01
7005 BN Doetinchem
The Netherlands
+31 (0)314 36 35 88
www.comfoor.com

University of Twente
Drienerlolaan 5
7522 NB Enschede
The Netherlands
+31(0)534899111
www.utwente.nl

Author
L.C.S. van Zadelhoff
University of Twente
Industrial Engineering and Management – Production and Logistic Management
lisanne_vanzadelhoff@hotmail.com

Graduation committee
Dr. P.C. Schuur
First Supervisor
University of Twente

Ir. W. de Kogel - Polak
Second Supervisor
University of Twente

Dhr. K. de Winkel
Company Supervisor
Operations Manager
Comfoor B.V.

Management summary

Comfoor is a company producing earmolds and earplugs. They sell them to dealers and industry. There are five main product types of earmolds and earplugs. These are normal earmolds (OS), LifeShell earmolds (LS), receiver in channel earmolds (RIC), acrylic earplugs (GHBA) and silicone earplugs (GHBS). The earmolds and earplugs are 3D printed and afterwards refined by hand to a comfortable and perfectly fitting product. Comfoor faces a changing customer expectation towards delivery time. Customers expect products to be available to them in a shorter time-frame than Comfoor currently realizes. Comfoor needs to decrease its process throughput time to be able to fulfil this market expectation. However, Comfoor has limited resources available. The desired throughput time is to produce within three workdays for at least 90% of all produced products. With this challenge in mind this research aims to answer the following research question:

"How can we substantially reduce the throughput time of producing customized earmolds and earplugs within Comfoor B.V.'s capabilities?"

From theory we decide to focus analysis on resource dedication (number of employees working on a certain production type) and cross-training (the ability of employees to work on a second product type). We analyze the production process and the time it takes to handle each process step. We use this information to build a discrete event simulation model. Next, we use a theory modelling approach to get a good starting point from which we derive experiments. From a time perspective we decide to focus our simulation experiments on the finishing production step. This is the step that needs most production time and is one of the steps that has the most resource restrictions. Therefore, the finishing production step has the highest potential of being the bottleneck. We use resource dedication and cross-training as interventions in the simulation model. Performance indicators such as service level, average waiting time and work in progress let us know which configurations will work the best. For the base setting of the simulation model we first analyze eight demand scenarios divided over the different periods of the year. From the average demand values of the demand scenarios we use a critical path method to see how long it would take to produce this average demand value. When infeasible, we add an additional resource to the process step that is the bottleneck on hourly throughput rate. Until we are able to meet this average demand on a daily basis within the three-day time limit. This way we balance the work-flows of the production process, making them more attuned to each other for a smooth work-flow.

With the starting position received from the critical path method we execute 8 x 32 experiments to determine better values for the number of employees needed. Afterwards we adjust the results that were not good enough with a greedy nearest neighbor heuristic. We define a good base setting for the whole year. Next, we improve upon this base setting by letting employees be able to handle two types of products (cross-training). When no products of the preferred type were present they would help the other product type. Afterwards we perform a paired t-test to be able to conclude on the results of the cross-training of employees.

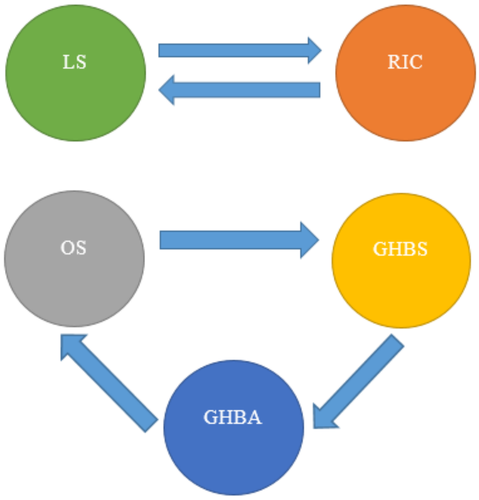


Figure MS.1. A nice way to implement cross-training

Table MS.1 below shows the base settings that work well over the whole year, except for June. This setting can be improved upon by letting employees specialize not only in one product type but in two. Every product type has at least one employee who knows to produce another product type. We made a presetting for this, depicted in figure MS.1. Because of the similarity in product type LS and RIC are linked with each other. The cross-trained employee that normally produces LS is also able to produce RIC. The cross-trained RIC employee is able to produce LS. Next, we linked OS, GHBS and GHBA in a rotating way. The cross-trained OS employee is able to produce GHBS, the cross-trained GHBS employee is able to produce GHBA and the cross-trained GHBA employee is able to produce OS.

	OS	LS	RIC	GHBA	GHBS	Total
# Full-time employees	4	2	2	3	4	15
Cross-training type	GHBS	RIC	LS	OS	GHBA	

Table MS.1 Final settings of the finishing production step

In June the performance of silicone earplugs decreases to an average throughput time of 4-5 workdays. However, this does not influence the total average throughput time much as the rest of the year this problem of excessive GHBS demand does not occur. Advised is to have extra capacity available in the June period when applying the same settings as described in table MS.1. Table MS.2 shows the average performance achieved with the settings of table MS.1.

Scenario: amount of cross-trained employee per product type	Average through-put time (workdays)	Average # WIP	Average service level	Average productivity finishers	Average waiting time products (hours)	Average workload waiting in the morning	Total number of stations occupied
1	2.23	3851	92%	98%	30.8	62%	15

Table MS.2 Performance measures for the final settings of the finishing productions step

We can conclude that the perceived seasonal influences do play a part but do not influence the amount of full-time employees that should be available daily. Flexibility can be created by making use of cross-trained employees. These are in a good position to start helping other product type departments when irregular fluctuations appear. However, extreme irregularities such as the high demand in June for the silicone earplugs are not fully coverable by making use of cross-training. Here extra capacity is needed to not exceed the required throughput time.

Further research should be done in the interaction with the modelling station and how many employees are needed there. Also, it is interesting to look at a good way to prioritize 3D print jobs on the 3D printers to ensure a good flow to the lab. The batch sizes by which the finishing employees finish the products can also be researched and a number of other throughput time reducing activities mentioned by Johnson (2003).

Abbreviations

OS:	Earmold
LS:	LifeShell
RIC:	Receiver in Channel
GHBA:	Acrylate earplug
GHBS:	Silicone earplug
A:	Arrival
R:	Registration
SN:	Cut
S:	3D Scan
ML:	Model LS
MR:	Model RIC
MO:	Model OS
MG:	Model GHBA
MF:	Model GHBS
P:	3D Print
C:	Clean
T:	Polishing machine
I:	Inject silicone
H:	Harden silicone material
PC:	Peel off cast
FL:	Finish LS
FR:	Finish RIC
FO:	Finish OS
FG:	Finish GHBA
FF:	Finish GHBS
EM:	Final check earmolds
EP:	Final check earplugs
S:	Ready to send
ET(i):	Early event time of node i
LT(i):	Late event time of node i
$E(T_{ij})$:	Expected processing time from departing from node i and arriving at node j.
$Var(T_{ij})$:	Variation on processing time from departing from node i and arriving at node j.

Table of contents

1. Introduction

In the framework of completing my master *Industrial Engineering and Management* with the specialization *Production and Logistic management* at the University of Twente, I performed research at Comfoor B.V. into the reduction of product throughput time.

For this research we used a simulation model to evaluate the performance of the production system under various scenarios and interventions. We compared these interventions to see which improve product throughput time the most.

In the next section we read about the company Comfoor B.V., from now on referred to as Comfoor. We dive more into depth on the desire to speed up the product throughput rate, the factors that influence increases in throughput time, followed by the research questions that guide this research and the scope of the project.

1.1 Introduction Comfoor B.V.

Comfoor is a make-to-order manufacturer of earplugs and earmolds situated in Doetinchem within the Netherlands. Founded in 1985 under a different name, Comfoor has grown and innovated itself into the company it is now. The biggest innovation is in 2004, the switch from conventional production to 3D printing by means of stereolithography. Comfoor employs 145 employees who cover approximately 100 full-time equivalents (FTE). Daily, about 1000 ear imprints arrive to be processed into customized products. The yearly revenue is 10 million euros.

Earplugs prevent hearing loss by reducing damaging sound levels. They are produced as Uni-Fit or Custom-Fit products and come in different shapes and sizes depending on the type of activity you want to use it for (see figure 1.1). No Custom-Fit earplug is the same, as every ear is unique.

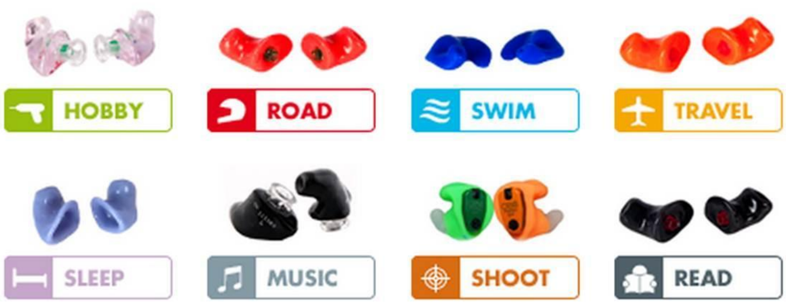


Figure 1.1. Different occasions to wear ear plugs

Earmolds are used to enhance sound and are connected to hearing aids. All earmolds are custom-made. There are different types to choose from depending on the form of the ear and preference in terms of visibility and firmness of the earmold. In figure 1.2 the most common variants are shown. The first one is the Bikini LifeShell, made so that it is almost invisible in the ear and has a minimal skin contact to the ear. The second one is a closed earpiece, which is made so that it closes of your ear entirely and it has a good fixed position in your ear. The third one is the receiver in channel (RIC). This earmold is nearly invisible in your ear. The receiver fits directly into the earmold, so it does not have to be fitted behind the ear.

Figure 1.2. The Bikini LifeShell, the closed earmold and the RIC

With this information about Comfoor and the products they produce we will now go into why a research into the product throughput time is necessary.

1.2 Reduction of product throughput time

Nowadays, the delivery time benchmark for companies that make-to-stock is: “order today get your product tomorrow”. Therefore, customers are more and more accustomed to receiving their orders the day after ordering. These developments also influence the expectation of the delivery time performance of make-to-order companies such as Comfoor. Market research at Comfoor confirms that customers first look to buy a Custom-Fit earplug, but after looking at the delivery time, revise their order to buy a Uni-Fit product instead. Custom-Fit product sales are lost this way.

This sentiment of not understanding why production has to take so long is shared amongst other customers. An audiologist, who visited Comfoor to receive training, stated during a tour around the facility: “Now I understand the amount of work it takes to produce a custom-made earmold or earplug. I have gained a better understanding of the amount of time I have to wait to receive it.”

Besides a business perspective there is also a human perspective to consider for the time-to-market of earmolds. An earmold is an extension of a hearing aid. Therefore, earmolds are a necessary caring product for people with hearing problems. A new earmold is typically ordered when hearing problems are already acting up. Therefore, having to wait therefore has a direct impact on the quality of life of the customer.

As market leader of earmolds and earplugs Comfoor’s main competitive advantage lies in being able to deliver products quicker and in higher volumes than competitors can. Currently, the production time for a custom-made earmold or earplug is 8 days on average. 94% of all produced earmolds are sent within 4 days and 90% of all produced earplugs are send within 12 days. These are figures of the first half of 2017. Comfoor wants to be able to guarantee delivery within 3 working days and preferably even shorter. Next to being able to deliver quickly, Comfoor wants to increase its market share. Currently, the product throughput time is too unpredictable. Because of this management is not certain how much demand it can attract in a short period of time and how this will affect production performance. Production performance is measured in being able to deliver within the agreements made with customers. For this research this means that sensitivity analyses will be made with regards to increased demand to evaluate the ability to fulfil these agreements.

So, we want to reduce throughput time to keep customers happy and gain more customers for the Custom-Fit products. But how will we find the answer? In the next section we propose the research questions which will guide us to the answers on how to reduce product throughput time.

1.3 Research questions and approach

How can product throughput time be reduced? This is the question we want an answer to. To find the answer in a structural manner, we will use the following main question during this research:

“How can we substantially reduce the throughput time of producing customized earmolds and earplugs within Comfoor’s capabilities?”

This question immediately leads to other questions. What is the product throughput time currently? What is the minimal throughput time for this production process? How can we achieve minimal throughput time? What are Comfoor’s boundaries? To answer these, we use the Ist-Soll approach. Let us evaluate the current situation (Ist) with the following questions:

Q1: “How is the production process organized currently?”

This research sub question will be answered in chapter two where we describe the current situation. This information is gathered by talking with the employees of Comfoor about their daily activities.

Q2: “What KPIs are currently in place and how does Comfoor score on them”?

Next, we are interested in current performance and its values as to better understand what we want to improve and can measure the increase in improvement later on.

After answering these questions, we know more about the current situation. However, we want to arrive to the Soll situation. How we want the future to be. The target is to produce products within three days with a minimal amount of resources. However, the way to arrive here is clouded by the complexity of the production process. We will unclutter the bottlenecks by using a simulation model and evaluate different scenarios and interventions. To uncover the bottlenecks that restrain us from arriving at the desired situation, we will answer the following questions:

Q3: “What does the simulation model have to look like to be able to answer our questions?”

This question will be answered in chapter four, where we will discuss the workings of the simulation model. The simulation model is a reflection of the system at Comfoor, described in chapter two. Moreover, the model has to be able to support the experiments we want to do, described in chapter 5.2. Finally, we want information on the performance measures discussed in chapter 5.1. Without calculating these performance measures we are not able to compare the different outcomes of different scenarios and/or interventions.

Q4: “What are the possible interventions Comfoor is able to implement?”

In chapter 5.2 we will further read about the scenarios and interventions considered. Interventions are adjustments in capacity and increasing flexibility of the workforce by means of cross-training.

Q5: “What are the restrictions Comfoor wants to work with?

Not all interventions considered will be financially interesting. A trade-off eventually will have to be made between extra costs and throughput time benefits. Costs do not have to be the only limiting factor, maybe certain theories are hard to apply in practice and do not yield the intended results. We will answer this question in chapter five, after which we know the initial results of the simulation study.

Q6: "What are the possible and probable scenarios Comfoor can find itself in the near future?"

In chapter three in which we will define the scope of the project. The scope limits the amount of scenarios we can look at. In a way that we can more into depth into one direction. As there are a lot of possible scenarios to consider, one must choose which ones probably have the highest relevance. The interesting scenarios are discussed and defined together with the operations manager of Comfoor. Another possible scenario is discussed with a team of employees and a consultant during a value stream mapping session.

Q7: "What is the sensitivity of the new-found model?

Lastly, we want to know how "future-proof" or how robust the results of this study are. Therefore, we have to change some parameters like error-percentage, amount of demand, different product mix to see how large the influence of these parameters is on the product throughput time. This question will also be answered in chapter five.

Now we know the influences different scenarios and interventions have on the performance of throughput time. With this information we can arrive to an advice on how to arrive to the desired situation (Soll) with the following question:

Q8: "What interventions are to be advised and how are these to be implemented?"

In question five we have our simulation model. With questions Q3, Q4, Q6 and Q7 we know what we want to experiment with in the simulation model. After executing these experiments, we find out which conditions will result in better performance. This can be transformed into a roadmap.

In the next chapter we acquaint ourselves with the production process of Comfoor, its product types, customers and priority rules.

2. Current production process Comfoor B.V.

First, we discuss the different product types Comfoor makes and the different customer types they sell them to. This because product type and customer type define the customer agreements made on the delivery time. In section 2.2 we read about the production process. Section 2.3 will show the priorities given over the whole process to certain product types. Lastly, section 2.4 elaborates on the current performance indicators that Comfoor uses to judge production performance.

2.1 Type of products and type of customers

Comfoor produces five major types of products which we consider in this thesis. Products with a total amount of less than 5% of the total production of 2016 are not considered for simplicity. The types are:

- Earmold (OS)
- Earmold – LifeShell (LS)
- Earmold – Receiver In Channel (RIC)
- Acrylate earplug (GHBA)
- Silicone earplug (GHBS)

There are also multiple types of customers these products can be sold to. Customer agreements on delivery due date depends on the customer type. Customer types are:

- Dealers
- PBM dealers
- Industry

PBM is a Dutch abbreviation that stands for personal protective equipment. PBM dealers are companies that sell personal protective equipment like acrylate or silicone custom-made earplugs. Dealers are shops where consumers can come by or make an appointment and buy their earmolds or earplugs. The delivery agreement is to deliver custom-made earmolds within a week and custom-made earplugs within two weeks. PBM dealers sell hearing plugs to companies. A delivery agreement of 3 weeks is made. Comfoor is also a PBM dealer itself. Industry stands for companies that need earplugs because of their loud working environment. Here a service is sold that employees keep having adequate protection for their ears. The internal agreement is to deliver within a month.

2.2 Current production process

Production process

The production of the different types of products is quite similar for the different product types. Every product follows the six steps depicted in figure 2.1 in linear fashion. The movie "Een kijkje achter de schermen bij Pluggerz!" on the facebook page of Pluggerz oordoppen shows the production process (Pluggerz oordoppen, 2017).

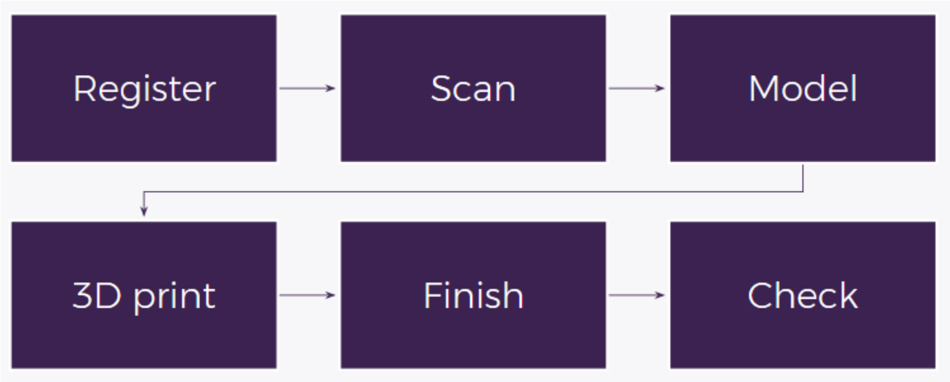


Figure 2.1. Production process, highly simplified.

Figure 2.2. Ear imprints.

First ear imprints are sent to Comfoor. In figure 2.2 we see a picture of some ear imprints. When these imprints arrive, the timer starts on production throughput time and the production process starts.

The first step in the production process is registration. At registration the product is marked as arrived and gets a sticker with which the product can be identified. The sticker gives personnel information on the product's requirements.

Next the ear imprints go to scanning. The imprint is scanned, by a machine, to make a digital image of it. Before the ear imprint can be scanned however, it needs to be cut to the right size. In such a way that only the usable parts are included in the image.

In modelling the digital imprints are transformed into digital earmolds or earplugs. Employees look at the digital image of the scan of the ear imprint and with help of software make a model for the desired product. When enough of these digital earmolds or earplugs are ready, a print job is made. The print job projects the model made for the final product in a way that the 3D printer can print them.

In the 3D printer the earmolds and earplugs are made. For silicon earmolds and earplugs the cast is 3Dprinted. When ready, these products are cleaned to get rid of the excessive fluids they contain. Clean earmolds and earplugs are hardened under UV light. In picture 2.3 below we show some examples of 3D printed products.

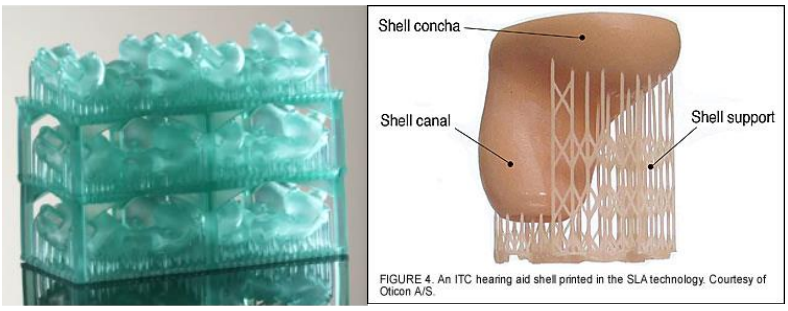


Figure 2.3. examples of how 3D printed products come out of the 3D printer

After the 3D prints are hardened the raw earmolds and earplugs need to be refined into end products. Here products go to different departments based on their type, because the finishing of the different types of products require different steps to finish. These different steps during the finishing are the following:

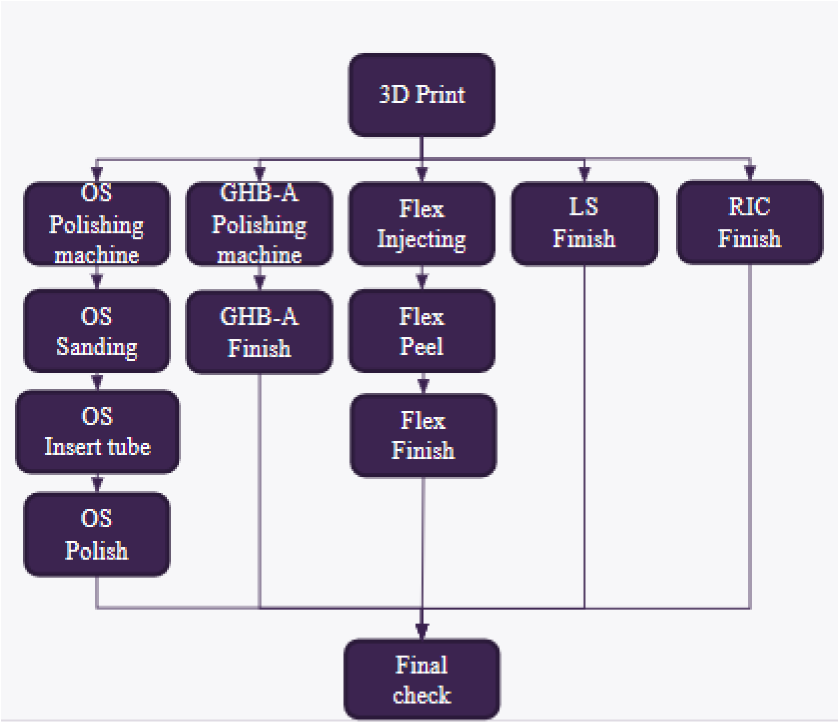


Figure 2.4. Finishing steps different product types.

The earmolds (OS) are first put in a polishing machine for an hour. Afterwards they are sorted on type and put in the polishing machine again for 15 minutes. During this process the unique identification code that is imprinted on the product during 3D printing is removed. Therefore, the products are colour-coded to be able to keep them apart. A coloured flexible tube is sticked into the vent, which is not affected by the ceramic stones within the polishing machine. However, there are only four colours available so eight customers receive the same colour code. Beforehand, eight customers are sought together that have different product specifications, so the products for different customers can be more easily kept apart just by looking at the product itself and comparing it to the specifications. When the earmolds are ready at the polishing machine, they need to be sanded by hand to refine the small areas the ceramic stones within the polishing machine cannot get to. A tube or nipple needs to be inserted that later connects the earmold to a hearing aid. Afterwards the earmold is polished for a nice finish. Every one of these steps happens on a different table at which employees have the necessary tools available. After one of the steps is done the batch of products that are finished are placed in the next buffer. Different persons execute the different production steps. GHB-A stands for acrylate earplugs and is first put in the polishing machine for one hour. Afterwards one person performs all the finishing steps for a batch of products. The silicone earplugs, in the picture referred to as Flex, first needs to inject the silicone material into the 3Dprinted cast. After which it goes into an oven to harden for 20 minutes. When cooled off the cast can be peeled off the silicone. These products are then ready to be finished. The injection, peeling and finishing is done in separate areas. Products are produced in batches of which the person that handles it has the responsibility. The LS and RIC are also handled in batches. One person produces one batch at a time. Production steps for LS and RIC products also include sanding by hand, adding additional parts, polishing the exterior with an anti-allergic layer of polish. In the end the products look like the ones shown in figure 1.1 and 1.2 of the introduction.

After finishing the products, they are brought to the final check. Good products are then made ready for transportation to the customer. Bad products re-enter the production process. About 10% of the daily products that are finished have to re-enter the production process. Depending on the error, they are rerouted to the appropriate production step.

To be able to see the whole process in one go we look at the flow chart on the next page. To be able to get the flow chart on one piece of paper the finishing steps are excluded. Because here the process splits as can be seen in figure 2.4.

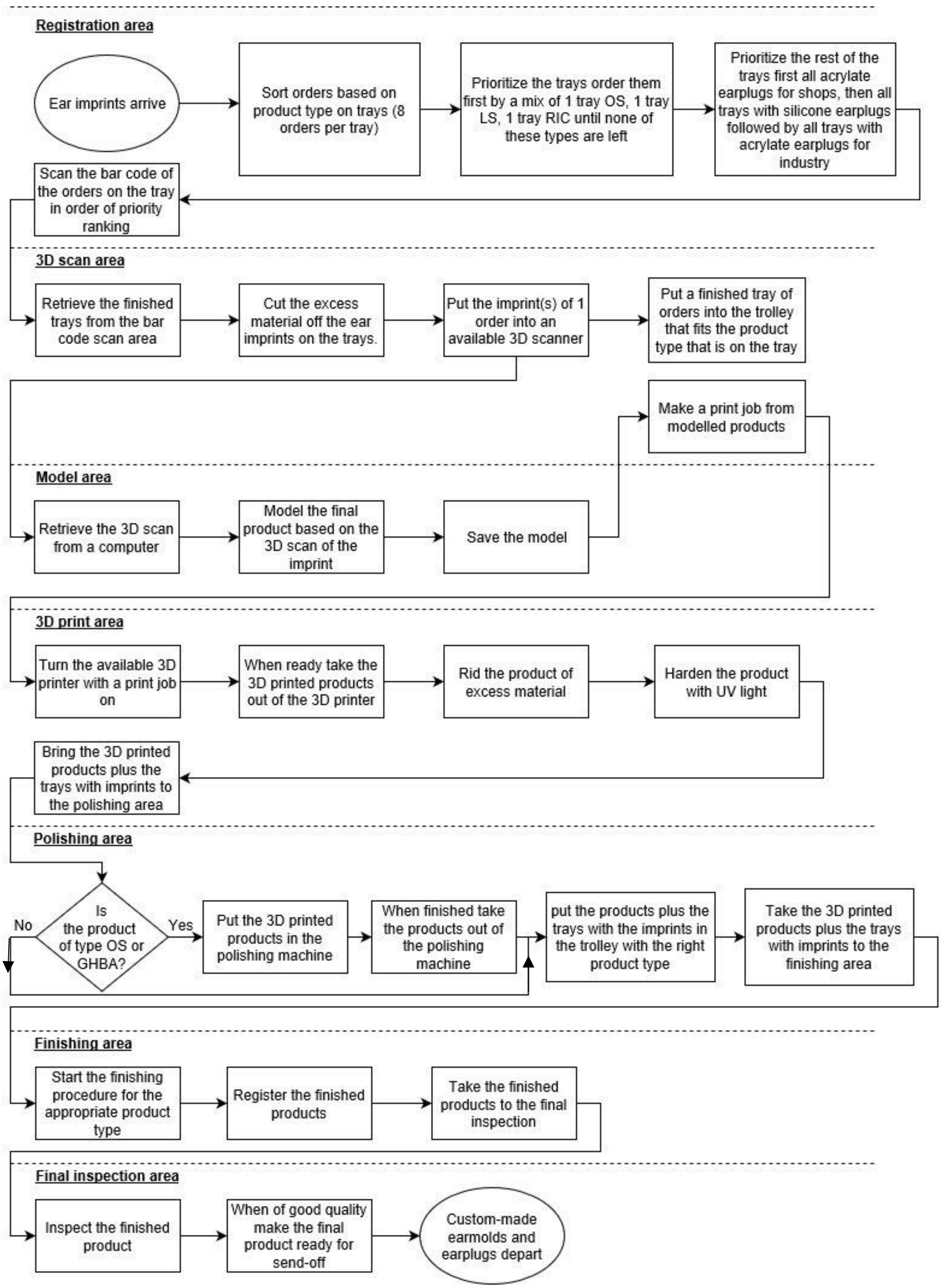


Figure 2.5. Flowchart of production process

2.3 Priority categories

Currently the products are sorted at registration according to product type. First earmolds, LifeShells and receiver in channel products are registered in a sequential manner per 16 at a time. After which the silicone and acrylate earplugs are registered. The modelling department first finishes earmolds, LifeShells and Receiver in Channels, mostly in the morning shift. After which the afternoon shift focusses on silicone and acrylate earplugs. Print jobs have the priority to first 3D print earmolds, then LifeShells and receiver in channels, afterwards silicone earplugs and lastly the acrylate earplugs. Everything but the acrylate earplugs is strived for to 3D print the same day the ear imprints arrived at Comfoor. The whole day someone is present to check earplugs and from 13:00 to 15:00 some employees join in on the final check to check the quality of the produced earmolds.

In the next section we look at the current way the production performance is measured.

2.4 Current key performance indicators

There are a few performance indicators in place that depict the status of the production process. These are used as guidelines to steer the company in the right direction. The foremost performance indicator is the average throughput time. This is supposed to be under three days for earmolds and under five days for earplugs. These restrictions are less strict than the ones we use in the paper, because we are trying to increase the performance in this area. Next there is the difference between incoming products and outgoing products. When more products go in than out than this is a bad sign. In the morning all products that are in the finishing area are counted. When work in progress arrives at higher heights than normal this is also an indicator that action needs to be taken. Furthermore, there are products that have been send back because the customer has a problem with them. These are guarantees. Its figure needs to be kept as low as possible. When the guarantees increase in number the employees are made aware. Performance indicators are intuitive, based on experience and change according to the situation within the year. With high demand, keeping track of strict deadlines is practically very difficult.

Now we know the current situation. In the next chapter we will research the possible actions to reduce throughput time from a literature perspective.

3. Theoretical framework

In this chapter we show why we chose for simulation. Next, we elaborate on the ways we could reduce throughput time and which ones we chose to implement in our simulation study. Furthermore, we chose a theoretical approach to fill in the production times and the starting position in the simulation model to compare interventions with. We read about this theory in section 3.3. Lastly, we define the scope of the research.

3.1 Simulation study

To analyse the production system and draw useful conclusions we need to choose in which framework we are best able to perform this within the available six months of research. Our options as Law (2015) stated in his book about simulation modelling, are to experiment with the system itself or make a model of the system. Experiments made with the actual system are mostly very disruptive and expensive. So, we choose to make a model. As it would be even more costly to recreate a physical form of the production system than to experiment with the actual system we opt for a mathematical model. Next, we need to determine if there is enough information and if the system is simple enough to use an analytical solution. Analytical solutions can provide optimal answers to our questions. But no, the production process is too complex, making the problem rather large for an analytical model. The additional advantage we see for using a simulation model is being able to add some different variations on the main model with relative ease and evaluate if there is a positive difference in using such a variation.

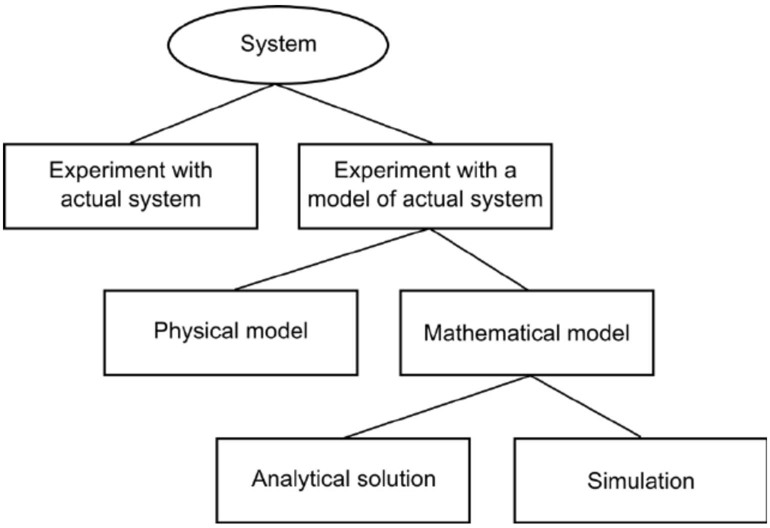


Figure 3.1. Ways to study a system from Law (2015)

We make use of discrete event simulation. The simulation changes the state of the system on predefined points in time. Such a change is then called an event. The time is kept track of by means of a simulation clock (Law, 2015). For example, a product is created at the start of the simulation, at time point 0:00. It immediately creates an event to move to the buffer for barcode scanning at the same time 0:00. Entering the buffer is an event which triggers a method. This method only works if the barcode scanner is operational and empty. When this is not the case the method creates a new event at 0:30 which executes the method again. Six and a half hours later on the simulation clock, at 6:30, the barcode scan becomes functional and the product is not blocked anymore. The next point in time that the method is called the product can be moved.

Next there is a difference between terminating and non-terminating simulations. With terminating simulations, the goal is to see the behaviour of a system in a certain period. In a non-terminating simulation, the steady-state behaviour of a system can be evaluated. We opt for a terminating simulation, because of the varying nature of demand there is no steady state throughput for the different “seasons” within a year. Therefore, we specifically want to research system performance in different circumstances. We make scenarios for these demand periods based on historical demand data.

Furthermore, we have to determine the starting position of the simulation. If normally the system is never empty, and we start with an empty system, the difference in output of the model compared to reality is too large and therefore unrealistic. Therefore, we have to calculate a certain warm-up period. For this period, we delete the data, so this data cannot “dilute” our calculations. We can also choose to fill the system beforehand with a certain number of products. However, we would have to determine for each scenario which number of products would be representable and change it whenever we want to run a different scenario. Therefore, we opt for the warm-up period method.

Due to the difficult and time-consuming nature of collecting data on production times we opt for a triangular approach using beta-functions. Beta-functions are commonly used in a PERT network where the durations to complete certain tasks are uncertain and hard to gather reliable data. Comfoor gathers data automatically in their ERP system, but it is cumbersome to transform the data into a suitable format. Furthermore, some of the gathered data in the system is unreliable because of the convergence of data from different systems which are not really compatible with one another. Beta-functions can take on a high variety of shapes and can therefore be a reasonable estimate of the real distribution. However, caution has to be taken when making use of this method. Because of a larger margin of error for using subjective measures instead of a high number of measures or objective data. Which can influence the outcome of this research. However, we are confident that we can at least give a good indication about which are good interventions to reduce throughput time.

3.2 Strategies for product throughput time reduction

We know we want to evaluate the performance in throughput time of the production system of Comfoor. However, we need to choose which parts of the production chain we would like to consider in our research and in what way we want to change to improve the system.

First let us define what exactly this system is we want to improve throughput time on. We can look as far as the whole supply chain or focus on the performance within Comfoor. Treville et al. (2003) asked themselves this same question, which one has more potency to achieve our goal, increasing demand information across the supply chain or increase production performance of the single company? They argued that to be fully agile, a company needs to receive full information on demand and to be able to fully observe demand. A company can fully observe demand when it knows the exact amount and type of demand to expect at least before production starts. The additional time between knowing demand and production can be used to set up or adjust planning.

Now let's take Comfoor. At day one there is a customer who wants to buy a custom-made earplug. He makes an appointment with the hearing care professional for day four and provides a full information transfer of his needs, a custom-made earplug. At the appointed date the hearing care professional makes an imprint of the ear. He makes up an order at Comfoor to produce the earplug. Afterwards he sends the ear imprint to Comfoor at the end of day five, because day four was really busy. Comfoor receives the ear imprint at day six and can start producing. In this case the hearing care professional has his information at day one, while Comfoor has partial information on day five. Comfoor will only know full information on day six when the ear imprint with specifications has arrived and production needs to start immediately, leaving no time to set up or adjust a planning. With full market mediation the hearing care professional could commit his demand for day six and Comfoor would also know demand at day one.

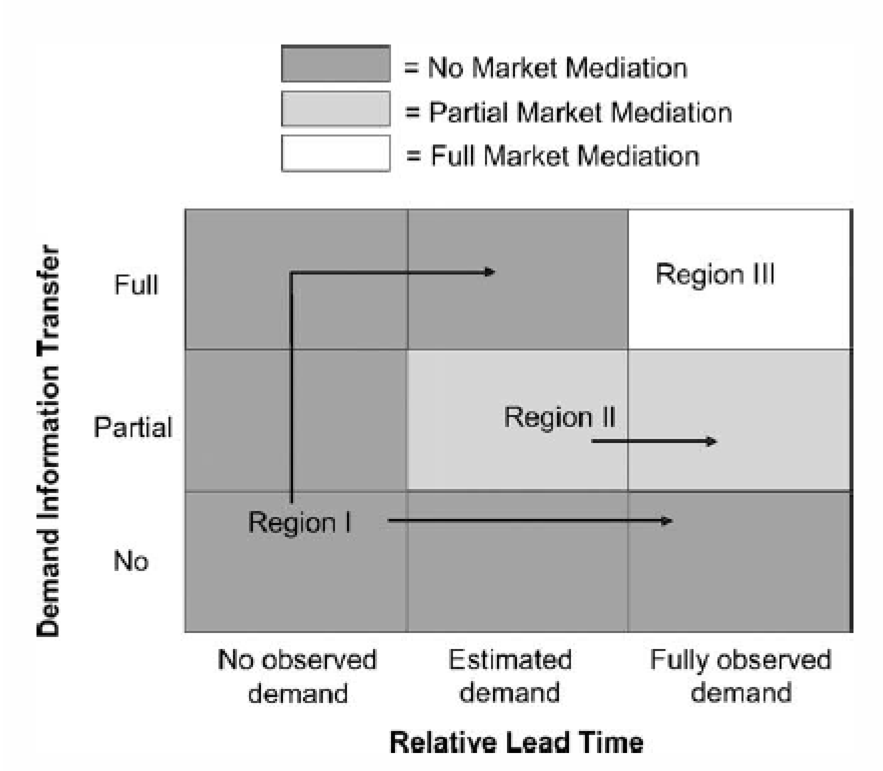


Figure 3.2. A demand chain typology from Treville et al. (2003)

Comfoor receives partial demand information from its customers. Demand is partially known when orders are put into the system. However, when demand is received to be able to start production is not certain. Demand can only be planned for once the imprints are in the company and they are all registered. At this time production starts. Looking at figure 3.2., Comfoor can be placed in the region: Partial demand information transfer and no observed demand relative to the lead time. In this case, Treville et al. (2003) advice to not focus on market mediation, but instead first increase supply performance. Increased supply performance can then increase reputation within the supply chain and allow for a position wherein market mediation is easier to achieve. Partial information is difficult to turn into usable information, therefore the strategy to come out of the region Comfoor is in is first to make sure all needed information is received that is necessary to plan and afterwards work on increasing the timeframe in which this information is received. Working on both discrepancies at the same time will most likely result in failure.

So, we do not focus on involving the supply chain in Comfoor's performance improvement, but first look at the throughput time performance in the company itself. Johnson (2003) developed a framework for managers to help breaking the available options down in parts, when looking at throughput time reduction. We see this framework on the next page in figure 3.3. There are four major strategies listed in column two: the reduction of setup-time, the reduction of processing time per part, the reduction of move-time per part and the reduction of waiting time per part. In the production process of Comfoor, a product spends most time in waiting. So, let us go more into depth into this category.

Moving further into the tree to column three that are ways to reduce waiting time per part, Johnson (2003) defined eight ways to reduce this waiting time per part. These eight ways can be broken down into five categories. Category one is reduce variability, category two is reduce batch sizes, category three is reduce utilisation of workstations and increase resource access, category four is reduce queues and category five is reduce setup time, processing time and/or move time. The reduction of batch sizes is only helpful when enabled with actions out of the last column of figure 3.3, depicted on the next page.

Hopp et al. (1990) also define five categories of strategies to reduce throughput time. These are very similar to the ones Johnson (2003) proposes. Category one, reduce WIP, is also a way to reduce total queue length. Category two, reduce batch size and transfer batch size to have queue control. Category three, synchronize production so that every station, before they start "overproducing", helps other stations. Category four, levelling work releases to receive a smooth work flow. Category five, eliminate variability.

While synchronizing production and levelling work releases are not present in the manufacturing throughput time per part reduction framework of Johnson (2003) they are strategies to reduce the total manufacturing throughput time. The rest of the categories is shared by both authors.

Now which ones are the most interesting in the situation of Comfoor? Variability in arrivals and processing time is difficult to evaluate and needs a lot of data and measurements. The arrivals are sorted on product type in the beginning and the employees from modelling determine the rate per product type that comes out of the 3D printer. The batch sizes are defined by the amount of products that fit into the 3D printer. The finishing step works with the same batch sizes except for the finishing of earmolds and silicone earplugs, which are produced per tray. Transfer batch sizes are at scanning based on how much a person can carry and the need for new products to process. Comfoor has two types of workstations. Machines and employees that have their own work space. The utilization of machines is mainly defined by the amount of time they are operating, which depends on the arrival of unprocessed parts and the number of machines available. Employee utilization is the same and depends on the number of employees working on the same type of production step and the arrival of unprocessed parts. Queues change from start of the morning at registration, scanning and modelling and finishing to the end of the day at 3D printing and finishing. The number of queues depend on the different steps that are defined as different process steps. For example, at the finishing of earmolds there are 4 different queues one before sanding, one after sanding and one before gluing in additional parts and one before applying polish. Meanwhile, finishing LifeShells has one queue before finishing, because all finishing steps are done by the same person sequentially.

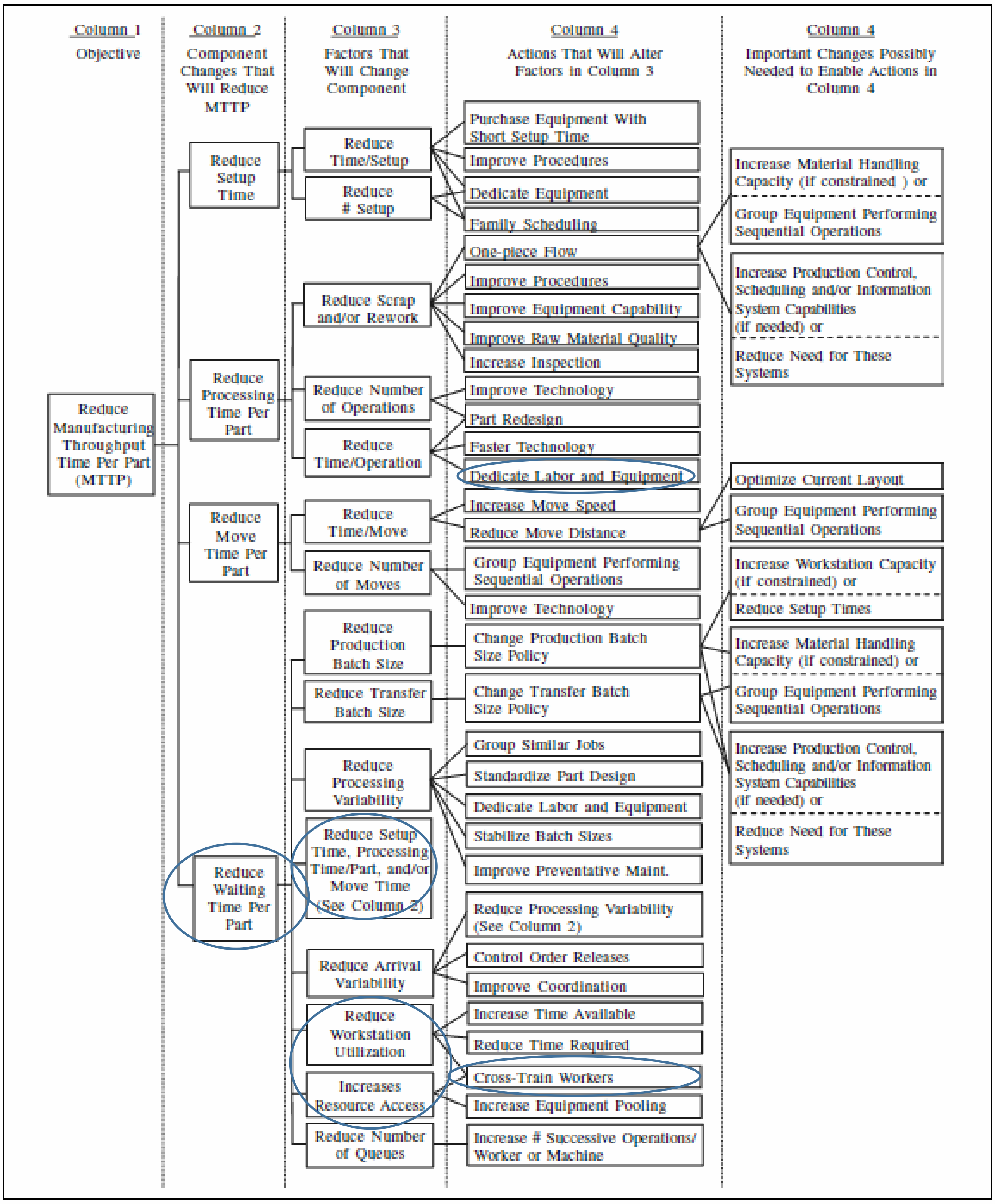


Figure 3.3. Manufacturing throughput time per part (MTTP) reduction framework from Johnson (2003).

Together with management we determined that the unknown demand with variable nature made the dedication of resources (labor and equipment in figure 3.3) very hard to manage. The large waiting times are often created by a misfit of resource dedication to demand. Therefore, we opt to research a good resource dedication in various scenarios, to make more visible what demand demands from Comfoor's resources. To make resource dedication more open to flexibility we also want to research to reduce workstation utilization and increase resource access by cross-training workers. Workers can be trained to specialize not only in one, but in multiple process steps. This way, cross-trained workers can change between their specialties and labour dedication becomes more flexible. This immediately defines a part of the scope. Now let us define the rest of the boundaries in which to perform this research.

3.4 Program Evaluation and Review Technique

To be able to validate the simulation model described in chapter four and arrive at a base state to derive our experiments off, we evaluate the performance of the current system with the program evaluation and review technique (PERT). Modelling, cleaning, injecting silicone, peeling off cast, finish and the final check are all production steps that are labour intensive. Therefore, process times have higher variation than when machines could do the task. Furthermore, data is difficult and highly time-consuming to gather manually, because not all needed data is collected automatically. Therefore, we chose to use the PERT approach. I asked employees, that do the job on a daily basis, how long it takes them to finish a product. How long it takes in the easiest of circumstances, how long it takes normally and how long it takes under the most strenuous of circumstances. This data can then be transformed into an average production time and a standard deviation. These times can be found in section 3.4.1. Adding times together and taking into consideration that products are produced in batches that fit within the 3D printers, we can look at the path products take and which of the paths are most critical. This we show in section 3.4.2. The duration of the most critical path will determine the total product throughput time. However, there are more products arriving in a day, they cannot be 3D printed at the same time (in the same batch). Therefore, we need to find the throughput time for processing multiple batches of products. This we show in section 3.4.3.

3.4.1 PERT production time estimates

Let us estimate the mean and variance of the different production steps by way of the PERT method. We have parameters a, m and b. a is the most favourable duration of a production step. m is the main duration of a production step, the so-called mode. And b is the least favourable duration of a production step (Winston, 2004). These parameters are estimated by subject-matter experts (SMEs), which are employees that deal daily with the appropriate production step. The expected mean of the duration of a production step can then be derived from the following formula:

$$E(T_{ij}) = \frac{a + 4m + b}{6} \qquad e(3.1)$$

And the variance on this mean from the formula following next:

$$varT_{ij} = \frac{(b-a)^2}{36} \qquad e(3.2)$$

The beta function needs two shape parameters: α_1 and α_2. We can use the above-named mean (μ) and the estimated a, b and m from an SME to get an estimation of these parameters over the interval [0,1] using the following formula:

$$\widetilde{\alpha_1} = \frac{(\mu - a)(2m - a - b)}{(m - \mu)(b - a)} \qquad e(3.3)$$

$$\widetilde{\alpha_2} = \frac{(b-\mu)\widetilde{\alpha_1}}{\mu - a} \qquad e(3.4)$$

With these formulas we assume that the data is skewed to the right. Law (2015) found that this is mostly the case when the object to be measured is a task time. So now we have an estimation for task time X ~ Beta(α_1, α_2) on the interval [0,1]. To transform this to an estimation for task time Y on the interval [a,b] we are looking for, we can use the following transformation:

$$Y = a + (b - a)X \qquad e(3.5)$$

Where X is the random variable obtained from Beta(α_1, α_2) (Law,2015). However, not all our data is skewed to the right. Some have a higher mean than mode, which makes them skewed to the left. Left skewed data does not adhere to the previous mentioned assumption. Therefore, we have to transform this data into a right skewed graph. We can do this by using Beta(α_2, α_1). However, to generate a representable random variable Y, we need the opposite value X provides us. Therefore, instead of using above mentioned formula we need the following formula:

$$Y = a + (b - a)(1 - X) \qquad e(3.6)$$

Because plant simulation uses real numbers with two decimals we round the parameters α_1 and α_2 to two decimals. Below we find the SME estimates and the estimates for α_1 and α_2 per process step:

Process step	a	m	b	Time-unit	$\widetilde{\alpha_1}$	$\widetilde{\alpha_2}$
Registration	8	10	15	sec	2.14	3.86
Cut ear imprint	15	30	50	sec	2.00	4.00
3D scan	90	120	180	sec	2.33	3.67
Model OS	4,5	5	6,5	min	2.00	4.00
Model LS	3,5	4	5	min	2.33	3.67
Model RIC	4,5	5	6,5	min	2.00	4.00
Model GHBA	2,5	3	4	min	2.33	3.67
Model GHBS	4	4,5	6,5	min	1.80	4.20
Cleaning	4	5	5,5	sec	2.33	3.67
Injecting silicone	27	37	95	sec	1.59	4.41
Peeling off cast	89	100	138	sec	1.90	4.10
Finish OS	7	8	9,5	min	2.60	3.40
Finish LS	7	11,5	13	min	4.00	2.00
Finish RIC	6	8,1	9,8	min	3.29	2.71
Finish GHBA	5,5	6	9,5	min	1.50	4.50
Finish GHBS	13	13,5	17	min	1.50	4.50
Final check earmolds	54	58	62.5	sec	2.88	3.12
Final check earplugs	31	41	56	sec	2.60	3.40
Ready to Send	12	14	19	sec	2.14	3.86

Table 3.1. SME estimates per process step and estimates for beta function parameters

For machine processing times like 3D printing and being polished in the polishing machine we assume deterministic processing times. For 3D scanning however not, because it's processing time depends heavily on the amount of material to be scanned.

3.4.2 PERT for one batch of earmolds (OS)

In sections 3.4.2 and 3.4.3 we will point to the activities using the following abbreviations:

A:	Arrival	C:	Clean
R:	Registration	T:	Polishing machine
SN:	Cut	FO:	Finish OS
S:	3D Scan	EM:	Final check earmolds
MO:	Model OS	S:	Ready to send
P:	3D Print		

We shall elaborate on one product type, namely earmolds (OS), to show how the calculations are done. We will first calculate the total throughput time of one batch of 20 earmolds. Second, we will look at the total throughput time of three batches of 20 earmolds to look at the increased complexity and explain the heuristic behind our calculations. Lastly, we calculate the total throughput time of average demand.

In section 3.4.1 we established the duration of each activity per product. However, Comfoor produces its products in batch form. This means that a product has to wait to go to the next production step until its “batch mates” are also ready with the production step. PERT makes the assumption that there are enough activities on the critical path to invoke the central limit theorem. We now assume that product one cannot continue until product 20 is also finished with the same production step, making the repetition of a production step part of the critical path. Therefore, we can sum up the mean of the duration of one activity 20 times to receive the expected duration of finishing 20 products in one production step. We can do this with the variation as well. The standard deviation of serving 20 products in one production step is the square root of the sum of variations. Producing earmolds is done in ten production steps. After arrival we register, cut ear imprints, scan ear imprints, model product, 3D print product, clean product, do the product in the polishing machine, finish the product, perform a final check, make the product ready for send-off after which the product can be delivered. However, each step is repeated for all 20 products. Registration first has to be done 20 times to start cutting the ear imprints. Therefore, there are 20 x 10 = 200 product handling moments inside the PERT figure depicted below:

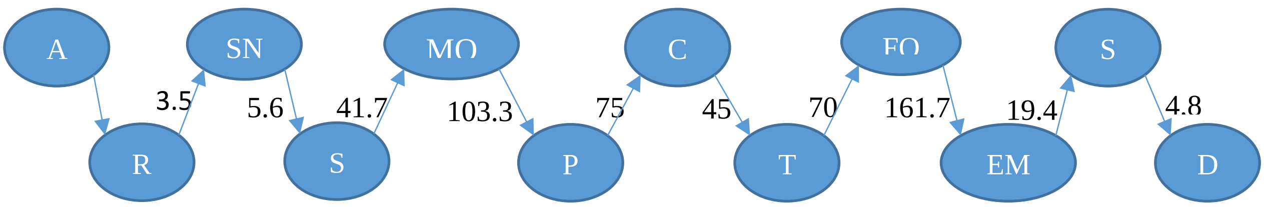


Figure 3.4. PERT for earmolds (OS).

From each activity we want to know which step needs to be finished before the activity can be started. Therefore, each activity has his own node (circle in the figure). In table 3.2 we see the immediate predecessor of each node and the immediate successor. These correspond to PERT figure 3.4. The early event time (ET(i)) is the earliest time the node activity can possibly start. For the first node this starting time is 0, as soon as production starts. Late event time (LT(i)) is the latest time an activity can start without delaying completion of production. Because every activity is on the critical path ET(i) = LT(i). Adding the expected duration and the variation of every activity 20 times we receive the following expected production times ($E(T_{ij})$) and variation on these expected production times ($Var(T_{ij})$):

Node	**Immediate predecessor**	**Immediate successor**	**ET(i)**	**LT(i)**	$E(T_{ij})$	$Var(T_{ij})$	**Starting time**
R	A	SN	0	0	3.5 min	0.5	6:30-7:00
SN	R	S	3.5	3.5	5.6 min	3.7	7:00
S	SN	MO	9.1	9.1	41.7 min	75	7:00
MO	S	P	50.8	50.8	103.3 min	133.3	6:00
P	MO	C	154.1	154.1	75 min	0	6:00
C	P	T	229.1	229.1	45 min	0	6:00
T	C	FO	274.1	274.1	70 min	0	6:00
FO	T	EM	344.1	344.1	161.7 min	208.3	7:00
EM	FO	S	505.8	505.8	19.4 min	0.7	13:00
S	EM	D	525.2	525.2	4.8 min	0.5	13:00

Table 3.2. PERT information of a batch of 20 earmolds (OS).

We can state that the total length of the critical path is 529.9 minutes. In hours this amounts to approximately 9 hours. The standard deviation is the square root of the sum of the variations: $\sqrt{0.5 + 3.7 + 75 + 133.3 + 0 + 0 + 0 + 208.3 + 0.7 + 0.5} = \sqrt{422} \approx 20.5$ minutes.

By means of the central limit theorem we assume that the distribution of the total time needed to create one batch of OS products is normally distributed with a mean of 8.8 hours and a standard deviation of 0.34 hours:

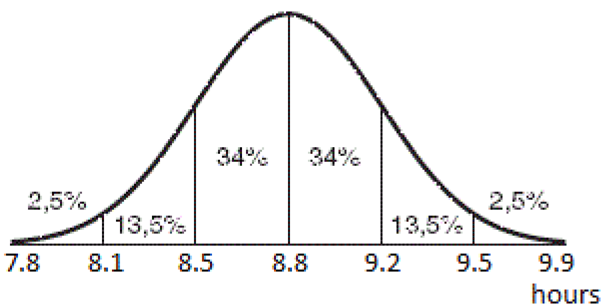


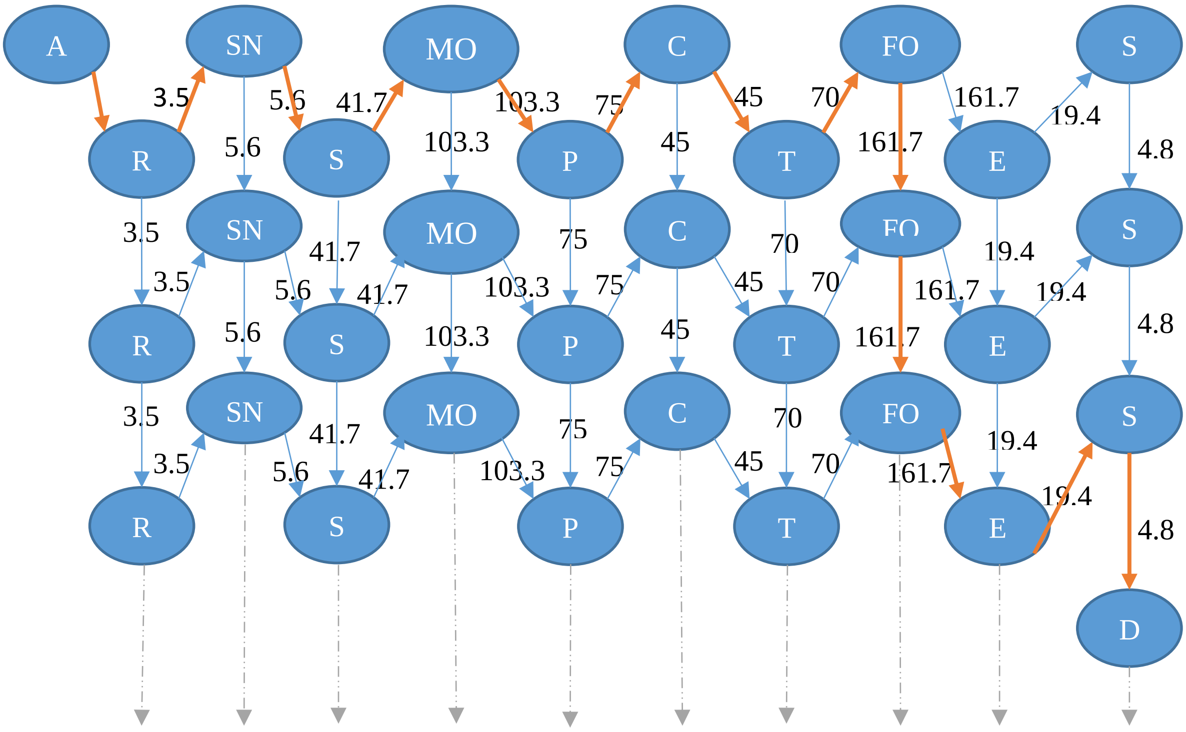
Figure 3.5. Normal distribution duration of producing a batch of 20 earmolds (OS).

The figure shows that there is a 34% chance that the duration of producing a batch of 20 earmolds lies within 8.5 and 8.8 hours. Also, there is a 68% chance that the duration of producing a batch of 20 earmolds lies within 8.5 and 9.2 hours.

The probability that this batch is finished before the three days are over is:

$$P\left(Z \leq \frac{24 - 8.8}{0.34}\right) = P(Z \leq 44{,}7) \approx 100\%$$

However, we are not really interested in how much time we can finish only one batch. We are interested how long it takes to finish the daily demand in products. For this we need to calculate the duration of an X number of batches. This can look like figure 3.6 when considering three batches.



# Paths	1	3	6	10	15	21	28	36	45	**55**

Figure 3.6. PERT for 3 batches of earmolds (OS)

To know the longest path between A and D we need only look to the activity that takes longest. All other activities can be done in equal or less the amount of time. The longest activity in above shown figure is FO. However, we do need to do the activities previous to FO at least once to be able to start the activity. The rest of the times that the previous activities need to be done can be done in the same time it takes for the longest activity to finish. After the FO of the last batch has finished we need to do the activities that follow also once for they could not start while the last batch still needed to finish step FO.

To add some numbers to this example: the total time it takes to go from A to D when finishing three batches of 20 OS products is 3.5 + 5.6 + 41.7 + 103.3 + 75 + 45 + 70 + 3 x 161.7 + 19.4 + 4.8 = 853.4 minutes. Adding the variation of these activities together: 0.5 + 3.7 + 75 + 133.3 + 0 + 0 + 0 + 3 x 208.3 + 0.7 + 0.5 = 838.6. Making the standard deviation, which is the square root of the total variation, of completing these activities 29 minutes.

Again, we'd like to know the chance the critical path has to finish three batches within three days.

$$P\left(Z \leq \frac{1440 - 853.4}{29}\right) = P(Z \leq 20.23) \approx 100\%$$

In a PERT model the chance that three batches are finished within three days is the sum of the product of the chances that a single path finishes within three days. The total amount of paths is 55. We can calculate this by looking per vertical row. Only including the first row, we can only define one path. Adding the second row to the first one gives us two extra paths. Adding the third row to the mix, we can define a total of six paths. When counting the possible paths while adding the fourth row gives us ten different paths. The numbers 1, 3, 6, 10 form a series together. Every time we go up a number we add one extra. To go from 1 to 3 you have to add 2. From 3 to 6 we add 3. From 6 to 10 we add 4. With this knowledge we add 5 to come to the next amount of paths which is 15, followed by +6, +7, +8, +9, and +10. In the end we come to a total of 55 paths as can be seen in figure 3.6. Assuming the other paths have about the same variation and are less in expected length the total chance of finishing the three batches within three days is approximately $100\%^{55} = 100\%$.

Until now every production step is accompanied by one machine and/or employee. When adding more products into additional batches, the production times increase to more than three days when keeping these settings. We need to add more capacity to be able to continuously be able to keep up with daily demand and not explode the system. We will discuss this in the next section.

3.4.3 Shifting bottleneck heuristic for a daily amount of earmolds (OS)

In this section we are going to use a simple version of the shifting bottleneck heuristic to make individual tasks smaller by adding more capacity to the activity that has the longest duration. By doing this, we increase the maximum throughput per hour so that hourly flow is more balanced throughout the process. We know we hit a desirable outcome when expected maximum capacity is equal or exceeds the daily demand we consider.

Let us take a daily amount of 240 earmolds (OS). This number is equal to the average amount of earmolds arriving daily at Comfoor. From this demand we can make 240 / 20 = 12 batches. Just as in the previous section we sum up the critical path. R + SN + S + MO + P + C + T + 12 x FO + E + S = 2308.3 minutes. The sum of the variation of duration of the activities is 2713.3. This makes the standard deviation 52.1 minutes. The expected duration amounts to 4.8 workdays of eight hours. We have a capacity of: one register, one cutter, one scanner, one modeller, one 3D printer, one cleaner, one polishing machine and one final checker who also makes the products send ready. From this moment on we call this the base capacity set-up. We assume that at least for the critical path no time is lost in changing between the production steps. Another assumption is that no products receive an error status and need to be remade.

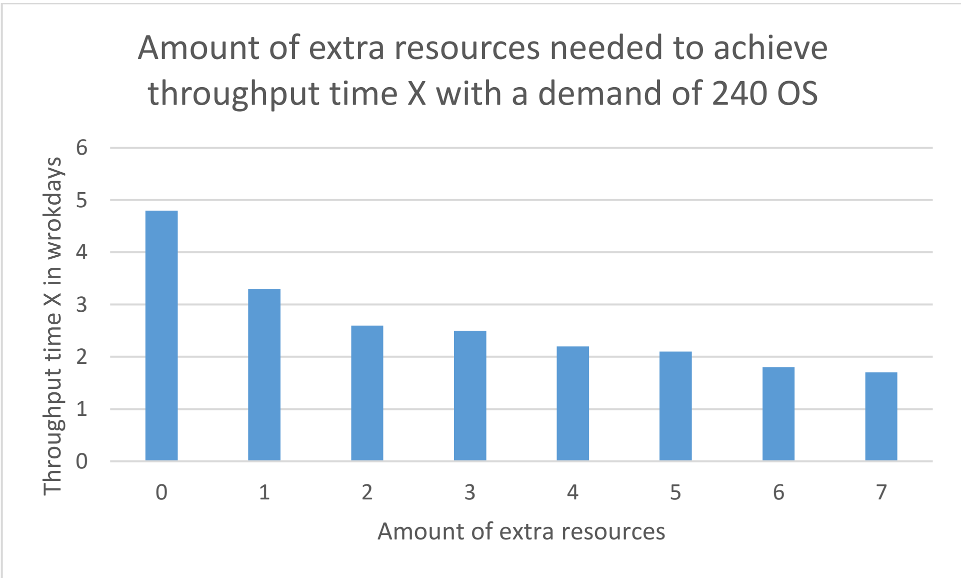
To reduce throughput time, we add a second finisher to the base capacity set-up. Therefore, we can divide the finishing time duration in half. Ideally, two finishers can divide the available products and work in parallel. Therefore, we change the finishing time to 80.9 minutes. The bottleneck, the activity that takes longest, changes and is now modelling, which still takes 103.3 minutes per batch. The critical path becomes: R + SN + S + 12 x MO + P + C + T + FO + E + S = 1585.8 minutes, equal to 3.3 workdays of eight hours with a standard deviation of 43.5 minutes.

This process continues in similar fashion. With the base capacity set-up, two extra modellers and two extra finishers the throughput time becomes 2.7 workdays. It reduces to 2.47 workdays with yet another extra finisher.

Only when the 3D printers become the bottleneck we cannot simply reduce the time by half. Because the production time is deterministic and does not depend on how many products are serviced. Therefore, we argue that when we want to increase capacity here we can produce double the number of products in the same time. We double the batch size and dedicate two 3D printers to the same batch. However, this also influences the activity duration of all other steps, for we now need to produce 40 earmolds (OS) per batch instead of 20 per batch. We double the original activity duration and divide this by the amount of resources we want to work on the activity. In numbers this looks like: 7 + 11.1 + 83.3 + 103.3 + 75 + 45 + 70 + 6 x 107.8 + 38.7 + 9.7 = 1089.8 minutes = 2.27 workdays.

This procedure can go on for some time. With every step we see that the amount of time won by allocating extra resources lessens and lessens. Therefore, we are interested in reducing the most throughput time with the least amount of added resources. Foremost because resources are expensive and will have to earn their weight in times of reduced throughput time. In graph 3.1 we can see the effect of adding resources according to the shifting bottleneck heuristic on the throughput time that we expect to achieve.

Let us make previous story visible:



Graph 3.1. Number of extra resources needed to achieve throughput time X

The extra resources added can be seen in table 3.3. Each step follows the previous step. For illustration, to come to step three and achieve a throughput time of 2.5 workdays we need: the base capacity set-up, one extra finisher from step one, one extra modeller from step two and one extra finisher for step three. We stop when we have reached a maximal daily throughput of the average daily demand of 240 earmolds.

# extra	Added resource	Total workdays	Standard deviation	Max # products daily
0	-	4.8	52.1 min	59
1	Finisher	3.3	43.5 min	119
2	Modeller	2.6	52.1 min	119
3	Finisher	2.5	20.5 min	178
4	3D printer	2.2	54.1 min	178
5	Finisher	2.1	46.7 min	238
6	Modeller	1.8	54.1 min	238
7	Finisher	1.7	29.0 min	297

Table 3.3. Results when adding extra resources to the base capacity set-up

We see in table 3.3 that for a demand of 240 OS a day we need seven extra resources to keep the throughput time up. We can approximate the distribution of throughput time by the following normal distribution:

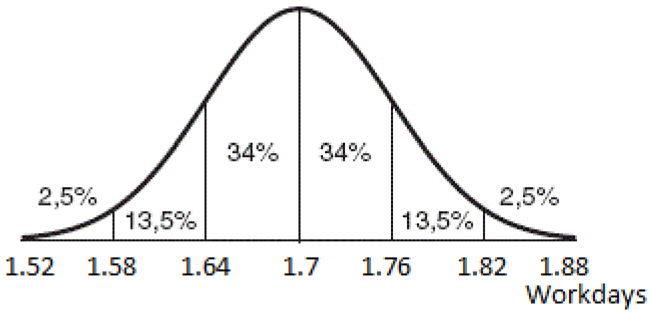


Figure 3.7. N(1.7,0.06) of 6 batches of 40 earmolds (OS) produced with 7 extra resources

How are our chances in finishing demand within three workdays?

$$P\left(Z \leq \frac{3 - 1.7}{0.06}\right) = P(Z \leq 21.67) \approx 100\%$$

Because the answer lies so close to one, the other paths have a high chance to also lie close to 100%. This way we can be confident that using the base setting with seven additional resources we added will make sure production is ready within three days and can do this on a daily basis.

To conclude, we estimate production times by means of beta functions derived from employee information. Next, we look at the daily demand of the entire year. We notice differences in demand patterns within the year and group similar ones together. We take the average of the time period of one group, which we call scenario. We divide the average daily demand by the amount of products that fit into a 3D printer and name this one batch. Next, we look if the makespan lies within our boundary of three days and if the available resources can fulfil the average demand of products within a day. If either one of these requirements is not met, we increase the capacity of the production step that takes the longest time to complete. We repeat the last two steps until both requirements are met.

Performing this heuristic on all product types we get the settings listed in appendix C. All numbers indicating the amount of resources needed is rounded up and measured in full-time employees that work eight hours a day. The heuristic indicates that the following amount of finishers, modellers and 3D printers should be present:

	Jan-Feb	Mar-Apr	May	June	July	August	Sept-Nov	Dec
# Finishers	19	19	17	17	15	15	19	17
# Modellers	10	10	9	9	8	8	9	9
# 3D printers	7	7	7	8	7	7	7	7

Table 3.4. Total capacity needs according to PERT heuristic.

The basis of production at Comfoor consists of 15 finishers, seven modellers and four 3D printers. However, in the first nine weeks of the year 2017 about 11.8 employees worked on average on finishing per day. In chapter five we will look if the simulation experiment gives the same base setting for production. Because we rounded up, the heuristic will require more employees than needed in reality.

3.5 Scope of the project

We already defined that our goal is to reduce throughput time within a limit of three working days by using labour and equipment dedication and cross-training, as explained in chapter 3.2. Plus, we are going to make use of simulation to research it, explained in chapter 3.1. However, with six big processing steps and per processing step multiple machines or employees to process products and five different product types, there are a lot of factors to experiment with. To keep this research within reasonable time limits we agree to only research the finishing step within the simulation model and treat the rest of the steps as given. The finishing step has the longest processing time when looking at the time it takes to finish a complete 3D printed batch and will most likely be the bottleneck of the process.

Now to come to a base state, we will use a heuristic based on the program evaluation and review technique the shifting bottleneck heuristic and line balancing. To see how this base state is performing we will look at its neighbouring solutions and evaluate the impact on performance. The explanation of the heuristic can be found in chapter 3.3. The precise definition of the experiments in chapter 5.2.

The goal will be to give a good indication of the influence of number of employees dedicated to finishing and the amount of cross training needed to achieve flexibility. The simulation model can later be used on a day to day basis to give insight in operational decisions concerning staff planning.

After this chapter we are a little wiser why we choose simulation modelling for this research, why we focus only on the process of Comfoor and why we choose the finishing step to focus on with the simulation study. Furthermore, we found a way to simply measure the amount of employees and machines needed to create a smooth product flow that is able to produce at least the average amount of daily demand. We can now head to the workings of the simulation model itself.

4. Simulation model

We create the simulation model in technomatrix plant simulation version 13.2. Our goal is to simulate the production process at Comfoor. We want our simulation to be a simplified version of the production process described in chapter two. With this simulation we can later evaluate the effect of changes made to the production process. So that, when the changes would be made in the real production system, we already have a good indication of what will happen.

We model the entire process of producing earmolds and earplugs as follows:

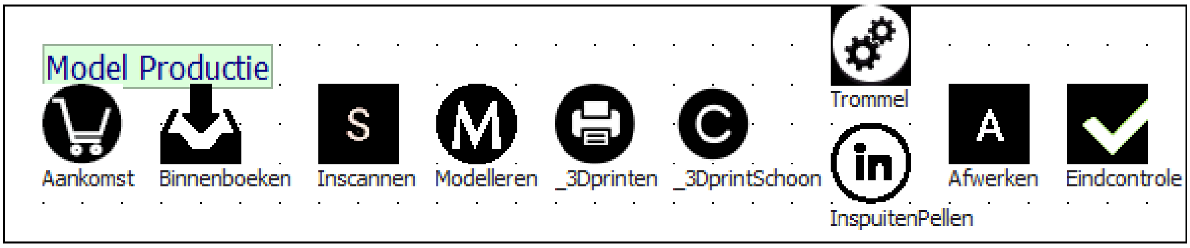


Figure 4.1. The production model

From left to right: first all products arrive in the morning at 0:00 at arrival. The next day the products are sent to registration. This day is not calculated in the throughput time, because it does not exist in reality. This additional time is build in to help the simulation run the way we want it to. At registration we choose an order of priority to start sending the products further. When registered, products go on to the 3D scanning module where they are cut and scanned. From 3D scanning they are sent to modelling. In the modelling station a predetermined number of employees is dedicated to a product type. When modelled, the products arrive at 3D printing. Here they wait to be batched and when enough products are ready they move on to cleaning. Depending on the product type the product moves on to either the polishing machine for OS and GHBA, injecting silicone and peeling of cast for GHBS or finishing for LS and RIC. After being ready at either the polishing machine or injecting silicone and peeling off cast the products move on to finishing. In finishing there will also be a predetermined number of employees with certain product type affinities. After products are nicely finished they move to the final check and from thereon out of the system. This flow is also depicted in figure 4.3 on page 27.

Not all production steps adhere to the same time schedule. Some production processes are worked on in shifts. A timeline of the day and the events that are time-bound are depicted in figure 4.2 on the next page. Registration starts at 6:30 and ends when finished or at 16:00. Scanning starts at 7:00 and ends when finished or at 19:00. Modelling has two shifts: one from 6:00 to 12:00 and one from 13:00 to 19:00. 3D printing is done in between 6:00 and 20:00. The polishing machine has the same operating times as the 3D printers. Injecting silicone happens from 6:00 to when it is finished or 16:00. Peeling off cast from 7:00 to 16:00 when there is work. Finishing happens from 7:00 to 16:00 as well as the final check of the earplugs. Earmolds are checked between 13:00 and 16:00.

Additional information on the simulation model can be found in detail in appendix A – a technical description of the simulation model.

We simulate a production day as follows:

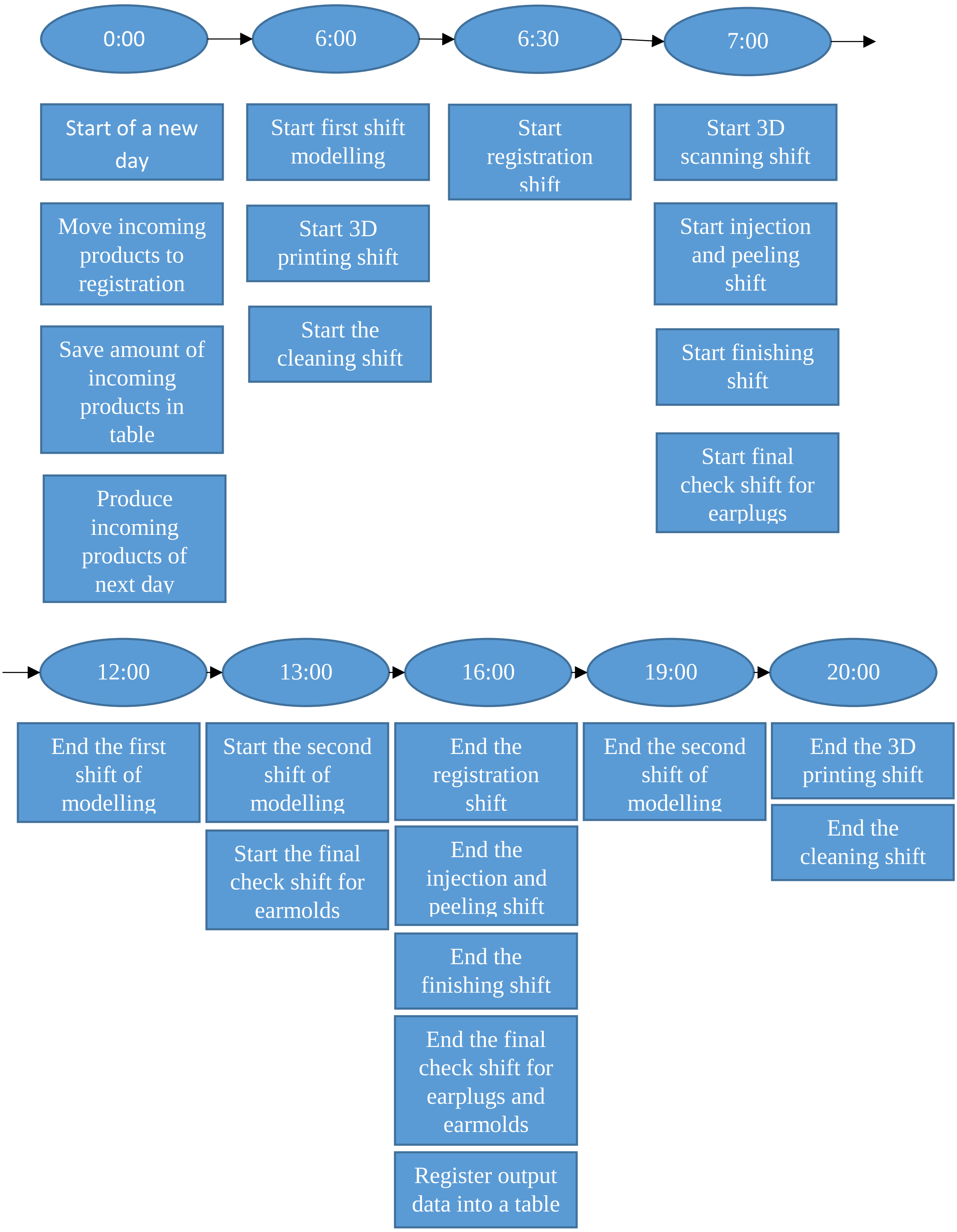
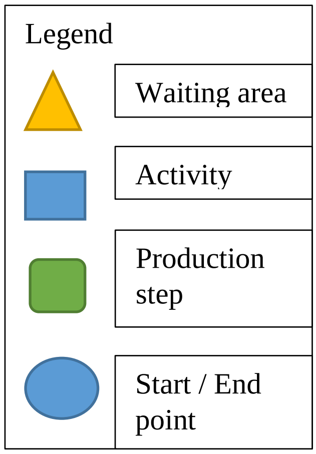


Figure 4.2. The production model time schedule.

The product flow develops as follows:



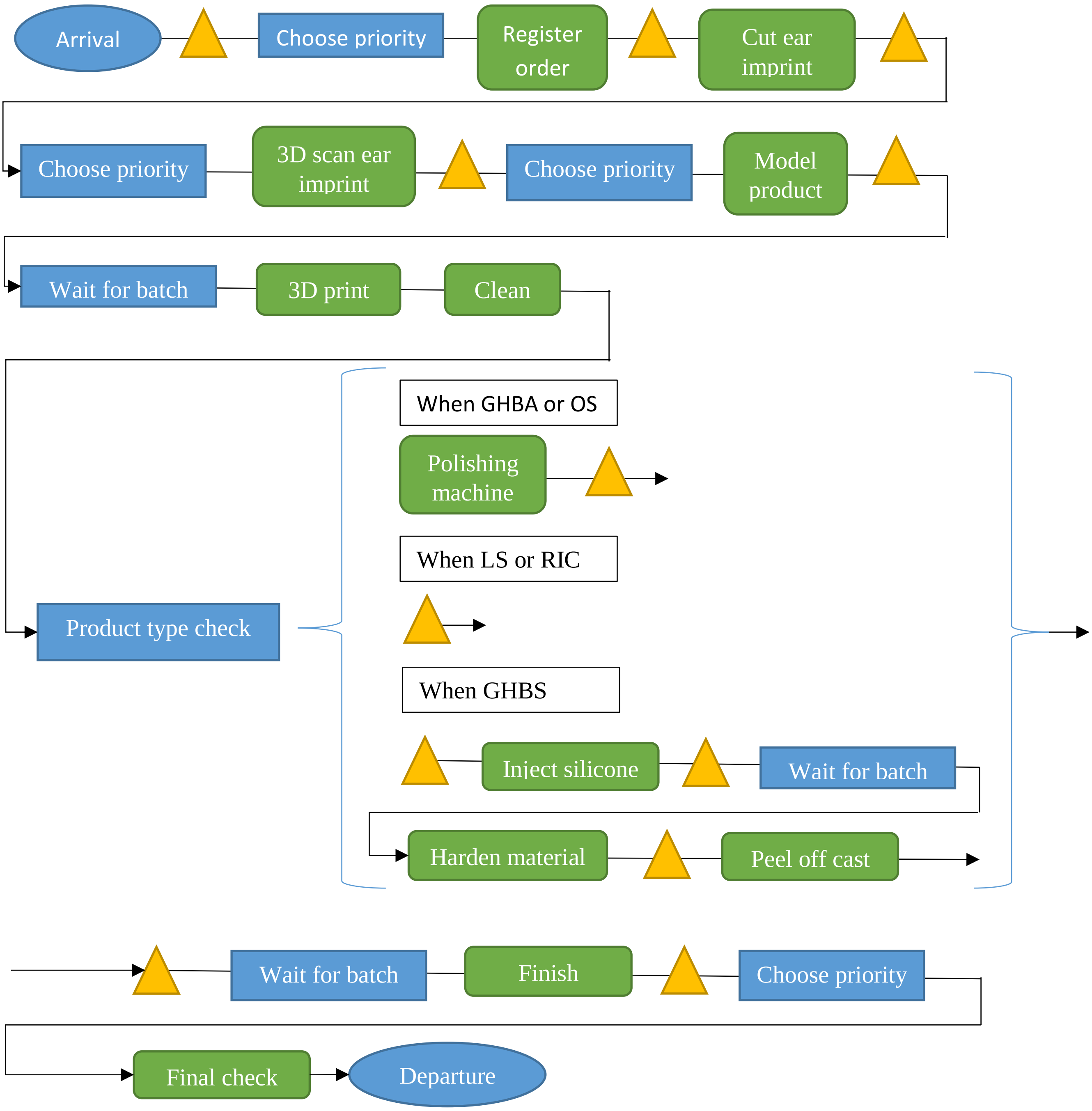


Figure 4.3. Production process simple flow chart

Assumptions

For the simulation model we make a few assumptions:

- Machines never break down and employees are never ill or unproductive.
- Walking distances between processes are non-existent -> cost no time.
- At the finishing step all different activities count as one: which means that they are performed in one go without interruption. One employee finishes one batch size (entered at the beginning of the project) at a time.
- 3D printers are made full or stay empty. Products waiting before the 3D printers can only be processed when at least the 3D printer can be filled entirely with one product type.
- Products processed in the 3D printer must be of the same type. No mix is possible.
- No overtime is possible.

Verification and validation

We verify the model by means of debugging. Next, we verify and validate the workings of the model with the operations manager.

Validation is further done by comparing the results of the PERT method calculations to the output of the simulation model. For example, when daily a deterministic demand of 240 OS comes in it takes on average 1.7 workdays to produce, this is confirmed by the simulation model.

Since we are using a non-terminating simulation model there will be no steady state. We can assume that a four-day warm-up period will suffice. Since our goal is to ready all the orders within three working days.

Inputs

In the simulation model we made a frame that supports choice making. Stations can be turned on and off, the 3D printer type can be chosen, the way products are handled at registration and a few other options. This frame can be seen in figure 4.4 below.

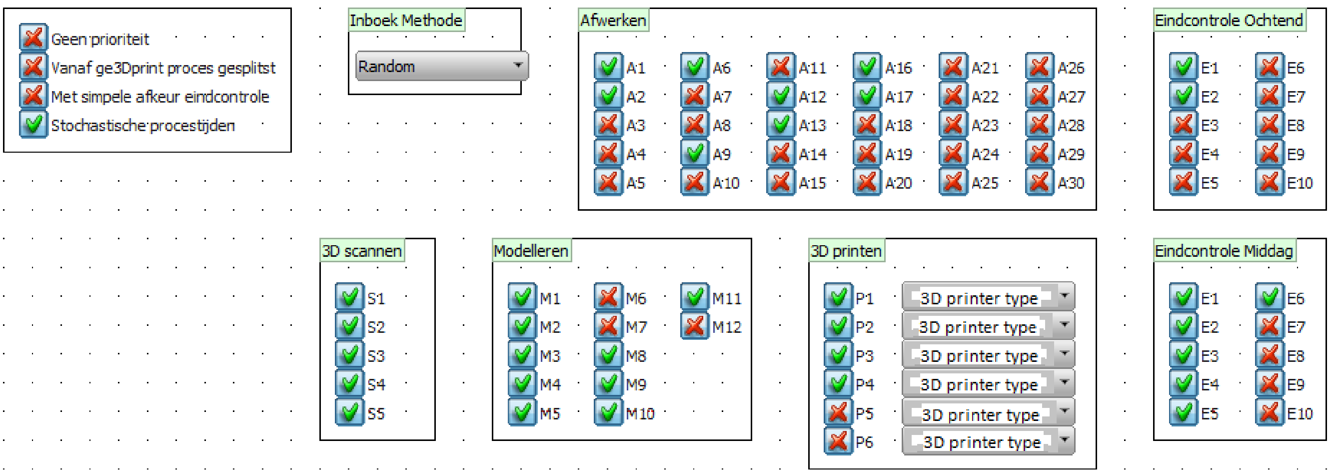


Figure 4.4. Input frame

For inputs we have on the left-hand side the options to evaluate the performance should there be no priority rules at all. The second option is to hold back products produced by the 3D printer until the end of the day, as per request incorporated. The third option is to cope with errors made internally. The last option is to calculate using distributions for the processing times.

The registration handling method can be one of four options:

- The current method, earmolds first in a rotating manner, followed by acrylic earplugs and lastly silicone earplugs.
- Silicone earplugs first, because it has an additional step in the process, more can go wrong, and production throughput time can be less certain.
- Random, which randomly chooses a product in the buffer to process.
- Equal priority, rotating between all types equally.

Next the amount of resources used can be chosen for 3D scanners (5 scanners), modelling (12 modellers), 3D printers (5 printers), finishing (30 finishers), final check morning shift (10 checkers) and Final check afternoon shift (10 checkers). There are different kinds of 3D printers that function differently. Therefore, the we can choose for each 3D printer in the simulation model what kind of 3D printer it is.

However, not every resource can handle any product type and there can also be prioritization in which type to handle first. That is why, we made it possible for every resource to have its own input settings which product types he/she or it can process, and which types have preference over others.

	Prioriteit 1	Prioriteit 2	Prioriteit 3	Prioriteit 4	Prioriteit 5	Prioriteit 6
S1	1	5	3	2	5	6
S2	2	3	1	4	5	6
S3	3	4	2	4	5	6
S4	4	1	3	2	5	6
S5	5	2	3	4	1	6

Figure 4.5. Possible priority and handling ability settings for 3D scanners.

So, every resource loops though the priorities from one to six. In the example of figure 4.5, S1 means 3D scanner one. The rest of the columns give the priority. From left to right the product types in the columns have priority over the ones that have a lower priority rank or are not even represented in the table. In figure 4.5 3D scanner one (S1) has product type one for priority one. Meaning that it will always choose to process an earmold (OS) first when there is one of this type of product waiting in the queue.

Finally, we need to input demand settings. We use historical data from 2017 and replay this in our simulation model. The data of the year that is not yet over we make an educated guess by comparing trends from 2015-2017. We distinct 8 demand scenarios: January – February, Mach – April, May, June, July, August, September-November and December. For these demand scenarios we calculate the average daily demand. Recall that with this data we could find a good point to start, from using the PERT method described in chapter 3.4, regarding necessary capacity. For January – February we see the findings below. The rest of the findings can be found in Appendix C – PERT settings demand analysis 2017. From top to bottom we have the months that cover the scenario and the different product types followed by the total amount summed over the product types. In the second row we see the average number of daily demand defined in amount of single products that arrive per day. Below is the number of employees needed on the finishing process step according to the heuristic of chapter 3.4. Next the number of employees needed on the modelling process step and finally the amount of 3D printers needed for a steady product flow according to the heuristic. When running the experiments we use actual demand data of 2017 or we multiply the data of 2016 with the current trend in increase or decrease of sales.

Jan-Feb	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average number of daily demand	245	95	125	210	235	910
# Finishers	5	3	3	3	5	19
# Modellers	3	1	2	1	3	10
# 3D printers	2	1	1	1	2	7

Table 4.1. PERT results demand 2017 January – February

Now we have a model that simulates Comfoor's production process and a starting point from which to start our experiments. We also know what possible changes we could experiment with. Let us define in the next chapters what experiments we choose to conduct, how we will evaluate their performance and what the results of these experiments are.

5. Simulation evaluation

Building a simulation is not enough to know more about the performance of Comfoor. We need to have performance measures, so we can compare outcomes of changes we experiment with. Furthermore, we need to set up experiments where we can be certain that we know what change we made and which effect it had on performance. In section 5.1 we elaborate on the performance measures we used to evaluate performance improvement or decrease. Section 5.2 describes the experimental design we set up and which changes we chose to evaluate. Section 5.3 provides us with the results of the experiments.

5.1 Performance measurement

To be able to see which scenarios and which interventions have higher performance and are therefore superior, we need some measure to compare them to each other. The main key performance indicator will be product throughput time and product throughput rate. **Product throughput time** can be measured in minutes, hours, days. This measurement resembles how much time a product has spent in the production process. In our simulation model we will calculate average throughput time in days. When the product has finished with the process, the amount of days the product resided in the system is written down. Off all products that were send that same day we compute the average of these amount of days.

Product throughput rate is the number of products that finish the production process per hour. Although we do not calculate it in the simulation model, we used it to determine needed resources. We calculated it by multiplying the hourly rate an employee or machine can make in an hour with the number of employees or machines that we use.

Work load is also an interesting performance indicator. It lets us see which department is short on capacity and which one has more air to breathe in between operations. Giving an indication to which reallocation or reprioritization would be helpful to look at. Because the finishing step is most time the bottleneck and resides in a later phase in the production process, this step has a lot of variety in arrival rate. As the factors that influence this are yet unknown to us, this is the most interesting step to calculate workload. Comfoor counts the amount of products that are at the finishing step in the morning before production starts. With this information we can see how much work is immediately available. Let us change the performance indicator to morning workload and define it as the percentage of work employees can immediately start with compared to what they can handle in a day. For example there are a 100 products lying in wait to be produced and per day 75 products can be produced with the amount of employees available. The morning workload would be 100 / 75 = 133%.

Productivity is the amount of time a resource is busy with processing products. In the simulation model it is calculated as being the number of products that are produced that day divided by the total amount of products that is expected to be produced when the resource devotes 100% of the time to processing.

The **target** is to have at least a throughput time of 3 working days. Every product that resides longer in the system is seen as late. An exception is made for acrylic earplugs for industry. These are determined internally and can suffer to have longer throughput times. However, it is business to make sure they do not stay in the system indefinitely. When we do not pay attention to this product type we will disrupt the system in a later time when the products are close to their due dates, letting the throughput time of the other product types suffer unnecessarily.

Service level is the percentage of products that are sent off within the target throughput time. Preferably we want this to be close to *90*%.

No costs are measured as the project should solely focus on achieving minimal throughput time with a minimal **amount of resources**.

In the following chapter we will acquaint ourselves with the ins and outs of the simulation model made for this research.

5.2 Scenarios, interventions and experimental design

Now we know what the simulation model can do we determine which scenarios and interventions we would like to run in the model.

We are running eight scenarios based on the average demand of each product type during this period. The average demand and the amount of finishers needed to produce this amount on a daily basis can be found in Appendix C. From each of these scenarios we choose 20 sequential days that formed the highest demand during that period. This to avoid planning capacity based on minimal necessity. We choose for the number 20 to make the simulation runs within the time limit. The model runs ten minutes per experiment. However, when running with 40 days the model runs 20 minutes per experiment, only running one instance. Therefore, we also choose not to use multiple runs to average out variation. This we look at later by evaluating the best options on their variation in output. Because of the high cost in time the simulation model takes, we limit ourselves to defining a good capacity option for finishing. Here we can use the results from the PERT analysis as input for the simulation model. Because during the PERT analysis we only stopped once we reached higher than average demand in our maximal capacity, we argue that probably we can use less finishers than we calculated.

For the first experiments we let every finisher only have one product type she can produce for simplicity. In the simulation model we can make use of a total of 30 finishing stations. The finishing stations A1 – A22 are used for this experiment, based from the PERT settings found in appendix C. They have all been assigned product types. A1-A5 produce earmolds (OS), A6-A8 produce LifeShells, A9-A11 produce Receiver in Channels, A12-A15 produce acrylic earplugs and A16-A22 produce silicone earplugs. We use the choices frame to select the stations we want to be on or off. The stations we want to experiment with we put into the experiment manager, which changes the values for each experiment.

To not let the amount of experiments go out of bounds, we choose for every product type to experiment with the last finisher needed. For example, if we calculated we need three finishers for Receiver in Channels we try it in the simulation model with three but also with two finishers. When we do this for every product type we come out on 32 different ways to combine variables. There are five product types and two options per product type which leads to a total amount of options of $2^5 = 32$. You can find the 32 combinations in experiments in Appendix D – Experiments.

It is possible that we do not outright have an acceptable solution. When this happens, we will make use of nearest neighbor search. Which the performance indicators within the simulation model we will take the product types which are underperforming and create settings with new experiments to try to improve on them.

From the results we will see if there is a setting that is able to cover each of the demand scenarios in an acceptable way. This would be preferable, because adjusting capacity is difficult. Learning the trade has a high learning curve and it takes about a year to be able to produce high quality with the necessary speed that is required.

The rest of the settings for the rest of the production process are put in such a way that they are not the limiting factor. We set the registering station to random processing. We work with five scanning stations. Ten Modelers will produce the digital prints for five standard 3D printers. All product types will have the same chances at being processed at the stations, except for acrylate earplugs for industry. The priorities that are applicable for scanners, modelers and 3D printers are depicted in figure 5.1. Modelers six through ten have the same priorities as modelers one through five.

	Prioriteit 2	Prioriteit 3	Prioriteit 4	Prioriteit 5	Prioriteit 6
S1	2	3	4	5	6
S2	3	4	5	1	6
S3	4	5	1	2	6
S4	5	1	2	3	6
S5	1	2	3	4	6

Figure 5.1. Priority rules for scanning, modelling and 3D printing

Furthermore, the final check will have four checkers for earplugs and six checkers for earmolds. Two of the earplug checkers will have acrylic earplugs as priority first and silicone earplugs as second priority. The other two will have the opposite priorities. The six checkers for earmolds have the three earmold types divided between them in the same way. So that every type has the same chance to be processed.

We run with a 10% chance at the final check that the product will have an error and has to be set back to the scanning department. At the end of the injecting silicone and peeling off cast step there is a 2% chance that the product has to return to the scanning department. The second set of experiments will be about cross-training employees. This topic is lightly introduced at the end of chapter 3.2. The objective is to create a more flexible workforce to be better able to respond to variable demand patterns. Employees will be able to not only finish one type of product but to be able to choose between two. However, we need to choose which second product type would be best to add to an employee's specialty. Recall from chapter 2.2 that finishing is labour intensive and includes sanding away sharp edges, gluing additional parts on and polishing the product to add an anti-allergy layer and make it good-looking. LifeShells and Receiver in Channel products both have a more delicate design than earplugs and earmolds. They also have a similar amount of daily demand. Therefore, we are grouping these products together. Earmolds have a pretty steady daily demand. Earplugs do not have a steady demand, here daily demand fluctuates most on a day to day basis. By rotating employees, as can be seen in the figure below, we can help equal out these fluctuations in demand. When one of the product types receives high demand, the connected type can jump in whenever they have time on their hands. First, we start by making one employee per product type able to handle the linked product type as depicted in figure 5.2. Next, we look if there will be improvement if two employees are able to handle two product types. We go on until all employees are flexible this way or we see little improvement.

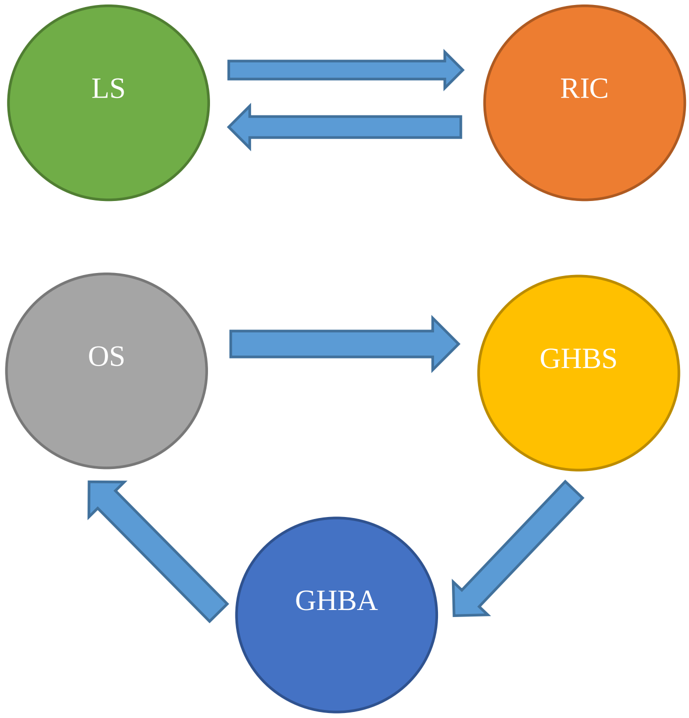


Figure 5.2. Employee knowledge rotation

5.3 Simulation results

We have run the experiments defined in section 5.2. Recall that the first 32 x 8 experiments we run the settings derived from the PERT method, described in appendix C, with the experimental design of appendix D. These experiments we do to know if the pre-set we calculated receives good performance. Furthermore, we can determine the influence of changing capacity at the finishing stations. The first run of experiments delivered the following best results seen in table 5.1. Results are the first setting that reaches 90% service level with the least amount of resources.

Scenario	Average through-put time (workdays)	Average # WIP	Average service level	Average productivity finishers	Average waiting time products (hours)	Average workload waiting in the morning	Total number of stations occupied	Experiment number (appendix D)
Jan-Feb	1.95	1448	93%	92%	23.6	36%	15	4
Mar-Apr	1.98	1593	93%	100%	24.7	42%	15	3
May	2.09	1530	93%	97%	26.8	48%	14	10
June	2.31	2421	81%	100%	32.7	134%	15	22
Jul y	2.55	2221	81%	100%	38.3	145%	13	22
August	1.95	1099	93%	87%	23.1	25%	12	13
Sept-Nov	2.18	1786	93%	100%	29.1	68%	14	32
Dec	2.33	1812	93%	100%	33.1	86%	13	2

Table 5.1. Performance of the 8 different scenarios and their respective settings

We rerun scenario June and July to look at which product type had the most morning workload. In both cases this turned out to be the acrylic earplug producing employees. This could have to do with the non-normal distribution of acrylic earplug demand in these periods, which make averages less trustworthy. We add an additional employee that can produce acrylic earplugs. The results can be seen in table 5.2.

Scenario	Average through-put time	Average # WIP	Average service level	Average productivity finishers	Average waiting time products (hours)	Average workload waiting in the morning	Total number of stations occupied	Plus station type
June	2.19	1975	92%	100%	29.7	36%	16	GHBA
July	2.25	1861	93%	100%	31.3	79%	14	GHBA

Table 5.2 New performance of scenarios June and July by adding one GHBA station

The best settings per product type that achieved the results from table 5.1 and 5.2 are described in table 10.3. The most left column shows the demand periods from which we derived our scenarios. The 6 columns that follow show how many full-time employees (working eight hours a day) we need to satisfy demand. In the last column we see the average throughput time in workdays (eight hours).

	OS	LS	RIC	GHBA	GHBS	Total	TH
Jan-Feb	4	2	2	3	4	15	1.95
Mar-Apr	4	2	2	3	4	15	1.98
May	3	2	2	3	4	14	2.09
June	3	2	2	3	6	16	2.19
July	3	2	2	3	4	14	2.16
August	3	1	2	2	4	12	1.95
Sept-Nov	4	2	2	3	3	14	2.18
December	3	2	2	3	3	13	2.33

Table 5.3. Preferred capacity settings within the finishing department

Below in table 5.4 we see the peak demand of 2017 for each product type. These are the extremes that need to be satisfied if we want to have a throughput time of less than 3 days every day.

	OS	LS	RIC	GHBA	GHBS
Max daily demand	280	126	184	420	495
# Needed hours of work	37.7	23.1	24.2	38.5	68
# Full-time employees	5	3	3	5	9

Table 5.4. Peak demand information 2017

However, opposed to just determining to take capacity equal to satisfy extremes, let's see what we can already mitigate by introducing cross-training. We take from every scenario from the first sets of experiments 2 weeks of demand information and string them together to create an artificial year of demand.

From above tables we get the following base setting if we sum up all needed employees per scenario and add the amount of needed employees for the peak demand. This sum we divide by nine to get the average need of employees (because we not only added the eight scenarios but also the peak demand information). We round to get a feasible solution. Only at GHBS we look differently, because in June there is a peak of 6 employees needed this increases the average number of employees in a way that we come to the number of 5. For most parts of the month this is unnecessary. Therefore, we stick here by the number that pops up the most, which is four.

	OS	LS	RIC	GHBA	GHBS	Total
# Full-time employees	4	2	2	3	4	15

Table 5.5. All year base settings finishing process

Comfoor has the following base setting for production;

	OS	LS	RIC	GHBA	GHBS	Total
# Full-time employees	5	2	2	2	4	15

Table 5.6. All year base settings finishing process Comfoor now

We see that it does not differ that much from one another. Comfoor slightly favours to have more capacity at the finishing of earmolds (OS) than on acrylic earplugs (GHBA).

Next, we want to create a flex pool with employees that know how to produce two product types. To be completely flexible we argue that every product type must at least have one person that could produce another product type if necessary. The simulation model is built in such a way that a cross-trained employee only handles the second priority product when there are not enough products in the buffer to handle the priority one type product.

Scenario: amount of cross-trained employee per product type	Average through-put time	Average # WIP	Average service level	Average productivity finishers	Average waiting time products (hours)	Average workload waiting in the morning	Total number of stations occupied
0	2.61	4121	83%	97%	39.4	95%	15
1	2.23	3851	92%	98%	30.8	62%	15
2	2.16	3758	93%	98%	29.1	54%	15
3	2.12	3769	93%	98%	28.2	50%	15

Table 5.7. Effect of cross-training employees

It is better to compare the results from table 5.7 with a paired t-test to see if the chance is significant or just happened by chance. In the first column of table 5.8 we see which rows of table 5.7 we are going to compare to one another. The N stands for the number of observations we have per scenario. The average difference, standard deviation and standard error speak for themselves. T stands for the t-statistic of the t-distribution. Next up we define a 95% confidence interval. The p-value is the chance that the average throughput times of both scenarios are equal. If this chance is under 2.5% the test is significant and we can say that we reject the statement that both throughput times are equal. The scenario with a higher amount of cross-trained employees has a lower average throughput time. The amount it decreased we can see with the confidence interval. The last column is if the test output is significant or not.

Comp.	N	Average difference	Standard deviation	Standard error	T	95% CI	p-value	Signific.
0-1	80	0.38	0.55	0.061	6.2	[-0.48,-0.28]	p < 0.0005	Yes
1-2	80	0.07	0.20	0.021	3.4	[-0.11,-0.04]	0.0005 < p < 0.001	Yes
2-3	80	0.04	0.17	0.018	2.2	[-0.07,-0.01]	0.01 < p < 0.025	Yes

Table 5.8. Paired t-test for cross-training employees, results of average throughput time performance

With a paired t-test we can compare the results with each other. All decreases in throughput time are significant. But the decrease quickly decreases in size. From having from every product type two employees who are cross-trained to from every type that has more or equal than three employees are cross-trained there is a 95% confidence interval that the decrease lies between five minutes an half an hour. This decrease is not relevant anymore and probably not worth the effort. The confidence interval is expressed in workdays, so we multiply with eight and 60 to get to minutes. We do not do the last experiment where every employee is cross-trained because we know from the last experiment that the increase in average throughput time is minimal.

Let us check the results by doing a paired t-test on the service level data of the same experiments. Here we can see how the reliability of the three-day deadline scores compared to the decrease in throughput time.

Comp.	N	Average difference	Standard deviation	Standard error	T	95% CI	p-value	Signific.
0-1	80	0.10	0.13	0.015	6.5	[0.07,0.12]	p < 0.0005	Yes
1-2	80	0.001	0.01	0.002	0.78	[-0.001,0.004]	0.2 < p < 0.25	No

Table 5.8. Paired t-test for cross-training employees, results of average service level performance

From having for every product type two employees who are cross-trained the service level does not increase. For the second configuration the only advantage is the decrease in average throughput time with an interval of [-52.8 min, -19.2 min]. This decrease is not really necessary as the average throughput time of setting one (where every product type has one employee who is cross-trained) is already well under three workdays (2.23 workdays).

The results of the experiments were interesting. There seems to be not so much variability in demand that a base setting for finishing is out of the question. Furthermore, we can improve upon the new-found base setting by implementing flexibility through cross-training. The effect however, decreases very quickly. Next chapter we elaborate on what we have learned from these results and have not yet learned.

6. Conclusion and future research

We chose in chapter three to study the production system of Comfoor by using simulation modelling. Then we developed a PERT method to validate the simulation model and to create a starting position to start the experiments with. After defining the experiments and elaborate on the results in chapter five we have now arrived at the conclusion and further research.

6.1 Conclusion

The main research question we want answer is:

"How can we substantially reduce the throughput time of producing customized earmolds and earplugs within Comfoor B.V.s capabilities?"

We can substantially reduce the throughput time of producing customized earmolds and especially customized earplugs, by dedicating our employees to the right product types. We can enhance performance by creating more flexibility in the dedication by having one person per product type group that knows how to handle an additional product type. When the product type the employee is dedicated to has less demand. The employee can choose to help with finishing other production types.

The base setting we advice to start staff planning from is having the following amount of full-time employees per product type as listed below in figure 6.1. The total amount of employees is exactly the same as in the base setting Comfoor uses currently. Except for the additional employee at finishing earmolds (OS) and one less employee to finish acrylic earplugs (GHBA).

	OS	LS	RIC	GHBA	GHBS	Total
# Full-time employees	4	2	2	3	4	15
Cross-training type	GHBS	RIC	LS	OS	GHBA	

Table 6.1 Final settings of the finishing production step.

In this case per product type one of the full-time employees is cross-trained into the cross-training type described in table 6.1. This will also help a lot to account for other discrepancies such as employee illness. There will always be at least one extra employee to employ on another product type if necessary. Unless of course multiple employees are ill.

At first, we thought that seasonal influences would make finding a good setting for the whole of the year impossible. However, the influence seems to only show in the extremes. There is a period in June where the average throughput time slightly exceeds three workdays, which occurs because the average throughput time for silicone earplugs reaches 4-5 workdays. For this period there is a need for at least one additional employee if Comfoor always wants to stay below three workdays. Otherwise the promotion on silicone earplugs should be thought over. For the rest of the product types only the silicone earplugs for industry reached sometimes above three workdays, because they are always treated last, but they never reached above five workdays.

Approaching the experiments with the PERT model created good first off results. They provided a slight overcapacity which we could reduce by means of the experiments we did afterwards. This saves a great amount of time to search for smart settings. However, the PERT method is not advisable to use as only method for calculating capacity needs, because it does not take into account efficiency in terms of use the least amount of resources that is necessary. Caution has to be taken as well, because when production times turn out to be slightly different, this could create different outcomes of the simulation model. However, we are confident that our outcomes are representative for the situation Comfoor is in now. However, should Comfoor grow or change considerably the simulation model should be used again to renew the base settings provided by this paper.

6.2 Future Research

Future research can be done regarding training of new employees. The time it takes for newly attracted employees to get the experience they need to function fully is very long and requires a lot of energy and time from experienced employees. It can be researched if training methods are available that let the employee learn herself, maybe video material, or practising on earplugs that will fit in her own ear, so she can partly evaluate her own work. Having shorter work-in periods can make the use of more flexible employees more interesting.

Another point is the reduction of rework. Currently rework can come up to 10% of total daily production due to new employees and new 3D printing solutions or issues. As every error increases production load reducing errors can significantly decrease delivery times. Especially individual production time but also mean production time. As illustration having 1000 inbound products of which 100 go wrong, next day also 1000 products come in and you have to process 1100 (when everything can be processed within a day).

Next it can be researched if some short-time boundaries can be set on the online ordering portal in a way that it becomes impossible to receive way more products than production can handle. An order is made online in an ordering portal. The date a product arrives depends on the time the customer sends it, the kind of delivery service and the country the customer is located in. Taking these factors into account it might be possible to determine the future system load with some accuracy. For Dutch customers perhaps a day and for Belgium customers about 3 days ahead. Boundaries could for example be set with a price difference per possible shipping day or the option to ship that day could be cancelled. An upper limit will help guarantee short delivery times to existing orders. Difference can be made to “preferred customers”. When for example Amplifon the Netherlands is a strategic partner, capacity can be reserved for this customer which makes it able to accept all its orders.

Another research area is the importance of delivery accuracy. Currently delivery is made within a range. Customers need to account for the “worst-case scenario”. This presumably lets customers plan their follow up appointment with their customers after this period of time, while the product may be ready sooner. Working within this agreement range also creates expectancies at customers that Comfoor is not always able to fulfil. Let us say the range is 5-8 workdays after receiving the ear imprints. In low demand periods Comfoor is able to keep to a five-workday lead time. However, when demand increases this lead time is hard to keep, leading in the worst-case scenario to 8 workdays. However, customers are accustomed to receiving their product after 5 workdays leaving them wondering why that is not possible this time.

Another possibility is to research how interesting it will be to really make the “Beter Horen” customer chain more of a strategic partner. Information exchange can help both companies with their planning.

Comfoor B.V. can benefit from knowing predefined appointments with regard to making a new ear imprints, or a reparation request, etc.. And the “Beter Horen” chain would benefit in more reliably knowing when they receive their products, so they can set their appointments on an earlier point in time.

Some orders are sent together but handled as individual orders while residing in the system. Linking these together will make sure that they can be planned to be ready at the same time. This way they can leave room for actual individual orders that do not have a prerequisite for sending. Researched can be how many orders are send together in which volumes. And how much time the first products in that order have to wait until the last ones are ready. Next a way has to be found to link these orders in an easy way to be able to plan the releases. Also, the benefit of planning releases to the work floor beforehand has to be researched.

Another possibility for future research is the ability to closer coordinate operations between custom-fit and uni-fit production. One of the disadvantages of custom-fit is the disability to quickly scale with demand. Nowadays there are large periods where the demand is relatively high, on average 1000+. When this is the case more employees are needed in the custom-fit department. During low demand periods there are either holidays or employees start to train in new competencies. When training lessons lessen because employees have adequate knowledge they are in the perfect position to help pre-process uni-fit products. This way operations can count on a maximum number of employees during high-demand periods while no working hours are lost during low-demand periods. Storage capacity and therefore costs to stock for uni-fits increase with this plan. Because during low-demand periods for custom-fit more uni-fits can be produced to satisfy demand over a larger period. However, because of the more stable nature of being able to supply orders that can be stocked beforehand uni-fit production is in a more comfortable position to vary its production rate. A negative point is that perhaps less people with disabilities can be hired, who are needed to adhere to regulations and which are a proud part of the company. Another issue is the willingness of employees to do “dumb” work. The production of uni-fits is a repetitious and relatively easy task.

Demand is season-bound this makes the amount of work highly variable. However, personnel have permanent contracts. An option would to look at the ability to use “min-max” contracts. With these contracts the company and the employee agree on a certain number of minimal hours that the employer has to accommodate to the employee. These hours the employee gets paid even if he does not work that amount of time. And on the other hand, a maximum number of hours that the employee is obligated to work in a week if the employer asks. This is not the only solution to achieve flexibility with a regular workforce. In the article “Tien technieken voor flexibele planning van vast personeel” there are 10 ideas to achieve more flexibility in planning capacity (Vieregge, 2010).

More research can be done with the simulation model in relationships between the amount of modelling stations available and the throughput rate this creates for the finishing step. Also, priority rules can be further investigated. Next there could be more research in the amount of time products wait between different product steps and what the reason is they have to wait. The strategies of Johnson (2003) could be used for this purpose. Lastly, the production times are really interesting. There could be more research in when down-times occur. Currently this is calculated within the production time, but to really know what time is valuable and what time should be eliminated as much as possible these different time categories should be measured apart from each other.

12. References

Hopp, W. J., Spearman, M. L., & Woodruff, D. L. (1990). Practical strategies for lead time reduction. *Manufacturing Review*, *3*(2), 78-84.

Johnson, D. J. (2003). A framework for reducing manufacturing throughput time. *Journal of manufacturing systems*, *22*(4), 283.

Law, A. M. (2015). *Simulation modeling and analysis* (5th ed.). McGraw-Hill Higher Education.

Pluggerz oordoppen. (2017, January 20). *Een kijkje achter de schermen bij Pluggerz!* [Video file]. Retrieved from https://www.facebook.com/pluggerz/videos/1272687139435279/

Vieregge, M. (2010, April 22). *Tien technieken voor flexibele planning van vast personeel*. Retrieved from http://www.logistiek.nl/carriere-mensen/blog/2010/4/tien-technieken-voor-flexibele-planning-van-vast-personeel-101132595

Winston, W. L. (2004). *Operations research: applications and algorithms* (4th ed.). Belmont: Thomson Brooks/Cole.

Appendix A – Technical Description of the Simulation Model

Main frame

In the following picture we can see the main frame for the simulation model. It has all the separate production steps, the performance measurement tables and the simulation variables which we can change for experimentation. The model is made with Dutch names for ease of use.

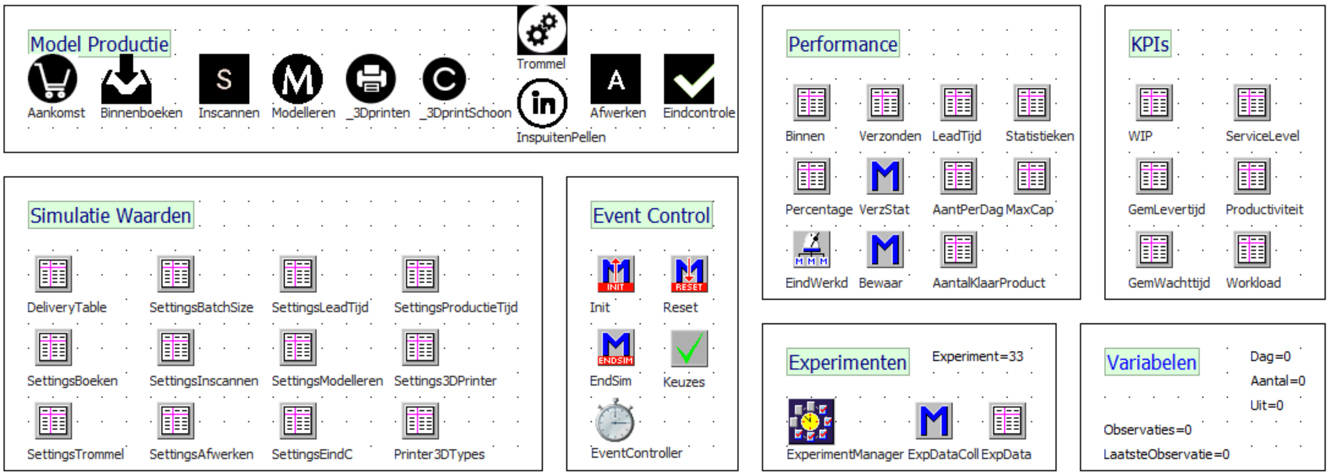


Figure A.1. Overall structure of the simulation model.

The main frame consists of 7 sub modules. At the top left we see the production model (Model Productie). Here we simulate the production logic. The process is cut into smaller activities to make the model more flexible and better understandable. Logic is applied differently for each production step as we will learn more about in the next section called production model.

Underneath we find the sub module simulation values (Simulatie waarden), which contains the input parameters of the model. In the delivery table for example we can input the number of products the source will produce. In the Settings for batch size we can determine how many products will form a batch together on certain steps in the process. With the table settings for lead time (SettingsLeadTijd) we can set the amount of days in which the products need to be ready for sending to the customer. In the table production times (SettingsProductieTijd) we set the production times of the products at the different steps in the process. For the rest we determine the number of machines and/or the number of employees that is available in the specified departments and which products they will be able to process, and in which order they will process it.

Next to the production model we have the sub module performance (Performance). Here we store the information we would like to know about the production process. The incoming products table (Binnen) shows how many products of which type enters the system. The outgoing products table (Verzonden) lets us see the number of products that are ready to be send away, in total and per product type. In the table number of products finished (AantalKlaarProduct) information is stored on all the products that entered the system on day x and which day they were ready. After a simulation run is finished this information is aggregated in the tables percentage (Percentage) and number of products finished per day (AantalKlaarDag). Which both show how many products that arrive at day x are finished at day y. The difference is that in the table percentage percentages are shown and in the table, number of products finished per day, amounts are shown. In the table statistics (Statistieken) information is shown on each product at what time it completed a step in the production process. The table lead time (LeadTijd) shows the total amount of time a product was in the system and if this time is more or is less than the set lead time demand. Lastly, the table maximum capacity (MaxCap) shows how many products modellers and finishers can produce daily, should they continuously have work they can do. Also, the maximum number of times the 3D printers can be turned on in total are shown here. The generator end of workday (EindWerkd) activates the methods store (Bewaar) and collect statistics (VerzStat) at 16:00. These methods fill in the performance tables and the key performance indicator tables.

Next to the sub module performance we see the sub module key performance indicators (KPIs). This module shows values on performance which we can use to evaluate the effect of changing settings. The table work in progress (WIP) shows the number of products that reside in the system. In the table service level (Service Level) the percentage of products that where ready within the set lead time demands are shown, in total and per product type. The table average lead time (GemLevertijd) shows the average time a product resides in the system. Next, we see the table productivity (Productiviteit). This table shows the percentage of time an employee is working. The average waiting time (GemWachttijd) is a table in which the average amount of time a product has to wait to be processed is shown. The last table is called workload (Workload) and registers the percentage of work there is in the finishing department in the morning compared to what can be produced within a day. For example, if there are 100 products at finishing and in total 50 per day can be produced, there is a workload of 200%.

The sub module event control (Event Control) initiates the simulation model with the correct values. It converts the information contained within the tables in the simulation model of the production. When a simulation is finished it collects the aggregate information in the tables showing performance. Plus, when hitting reset it deletes the appropriate data within the model. Choices (Keuzes) is a frame in which we can change settings for the model, such as turning stations on or off. Lastly event control also contains the event controller which monitors time and starts events.

Underneath the sub module performance, we see the sub module experiments (Experimenten). This is the experiments tab. It contains a counter to keep track which experiment is currently run. Next it also contains an experiment manager with which experiments can be formulated and started.

In the bottom right corner, we see the sub module variables (Variabelen). The variable amount (Aantal) counts the number of products that arrive per day. The variable out (Uit) keeps track of how many products are ready for send-off per day. The variable observations (Observaties) shows how many products have entered the system in total. The variable last observation (LaatsteObservatie) shows how many products have left the system in total. The variable day (Dag) shows the current day simulated within the model.

In the following chapters we will go into more detail in the different modules that make up the production model.

Production Model
The production model strives to represent the current production process at Comfoor B.V. and possible future improvements on this process. For example, during the process of this thesis a few older model 3D printers were sold and a new model was bought to replace them. With this new model the production process already changes. As we can see in figure X from last chapter the production model exists of:

- Arrival (Aankomst) which simulates the arrival process of to be produced orders.
- Registration (Binnenboeken) which simulates the registration process.
- 3D scanning (Inscannen) which simulates the scanning process.
- Modelling (Modelleren) which simulates the modelling process.
- 3D printing (_3Dprinten) which simulates the 3D printing part of the process.
- Cleaning 3D prints (_3DprintSchoon) which simulates cleaning of the 3D printed products.
- Polishing machine (Trommel) which simulates the sanding in a polishing machine part of production.
- Injecting silicone and peeling off cast (InspuitenPellen) which simulates the injection of silicone into cast prints and the peeling off of the cast prints once the material is hardened.
- Finishing (Afwerken) which simulates the finishing part of production.
- Final check (Eindcontrole) which simulates the final check and packing part of the production process.

How these different modules are built up will follow in the next subchapters.

Arrivals
The first module is to simulate the arrival (Aankomst) of products at Comfoor B.V.. Each day products arrive before production starts and are send to registration.

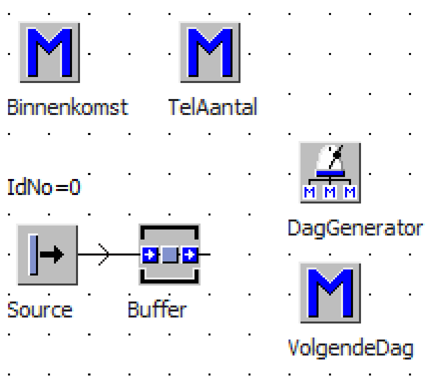


Figure A.2. Arrival process

In the model we simulate the arrival process with the source. Every day at 5:00 the source creates the number of products that is put inside the delivery table on the main frame. When a product is created the method arrival (Binnenkomst) will give it an unique identifier and the correct production times depending on the product type. With every product created the variable IdNo will go up with one. The unique identifier is a combination of the next day, the product type and IdNo. [Currently production times are the average values provided by Comfoor B.V..] After the method arrival is ready it calls on the method count amount (TelAantal) to increase the counter amount (Aantal) and observations (Observaties) on the main frame by 1. At 0:00 the day generator (DagGenerator) generates the next day by starting the method next day (VolgendeDag) which increases the day (Dag) counter on the main frame. The variable amount is set back to zero, after filling in its value in the table incoming products (Binnen) as total amount of products received the next day. Additionally, all products in the buffer (Buffer) are moved to the next module registration (Binnenboeken). Because the final check has two different shifts that depend on the time of day the method next day also resets the number of employees that is working the final check back to its morning values.

Registration

The registration (Binnenboeken) process is the first process by which we can start differentiating in the way we process products.

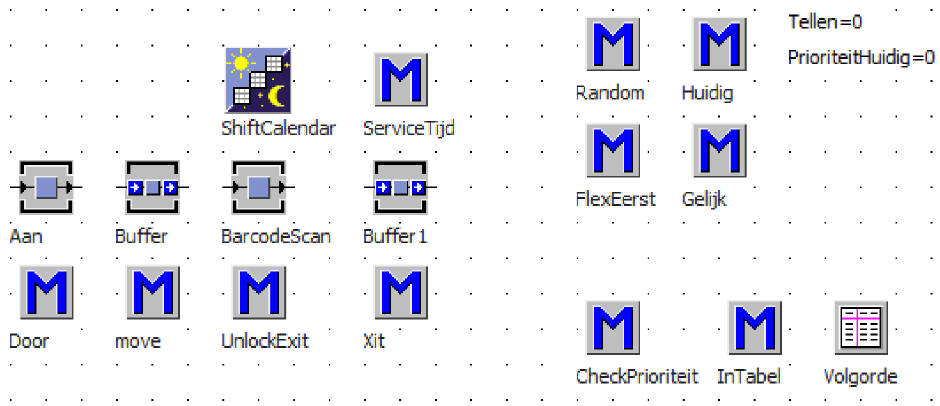


Figure A.3. Registration process

At the start of each new day the products that arrived previous day will be transferred to the incoming station (Aan) which will store them in the buffer (Buffer) unit by means of the method "Door". The method move (move) lets the products wait in the buffer until the employee that registers has his hands free. Depending on the selected method of registering that can be chosen in the table settings for registration (Settingsboeken) on the main frame we check which product has priority or not. For simplicity four different methods are available here.

- To register randomly without sorting represented by the method random (Random).
- To register the earmolds (OS), Lifeshells (LS) and receiver in channels (RIC) sequentially per 16 products after which all acrylate earplugs for shops is registered an afterwards all silicone earplugs, followed by all acrylate earplugs for industry. This is represented by the method called current (Huidig) because it is the current way of registering.
- To register all silicone earplugs first and afterwards apply the previous option to sequentially register OS, LS and RIC's, represented by the method silicone first (FlexEerst).
- To register all products except for the acrylate earplugs for industry sequentially per 16 products, represented by the method equal (Gelijk).

When the method random is chosen, priority is not important, so it does not have to be checked by the method check priority (CheckPrioriteit) nor is it needed to store products in the table order (Volgorde). These methods only apply when one of the other three registration methods is chosen. An error handler is built in in the initializing (Init) method on the main frame so only one registration method can be chosen during a simulation run. The variable counter (Tellen) counts the number of products that have passed with the current priority type. The counter resets if the priority, kept track of by the other variable current priority (PrioriteitHuidig), is changed. The method put it in your table (InTabel) stores all products that enter the buffer inside the table order until they are moved to register (BarcodeScan). The check priority method looks at the counter for the current priority and looks if already 16 products of that type have passed. If not and there are still products of that type in the buffer it moves the top product of that type listed in the table order to the employee at the register, removes the product from the list and increases the counter counter by one. If there are no products of the current type in the buffer the priority is changed according to the chosen registration method also when 16 products have passed the priority is updated according to this chosen registration method. The time the employee at the register is available is listed in the calendar (ShiftCalendar) which is from 6:30 – 16:00. When products are ready they arrive in buffer (Buffer1) after which they are moved to the scan (Inscannen) module with method exit (Xit). This method also registers the time at which the product has finished registration inside the user defined attribute registered (Ingeboekt).

Scan

Below in Figure A.4 we can see the contents of the module scan (Inscannen). After registration a digital print needs to be made from the ear imprints (which are a physical representation of a customer's inner ear). From this digital print the modellers can later design the product.

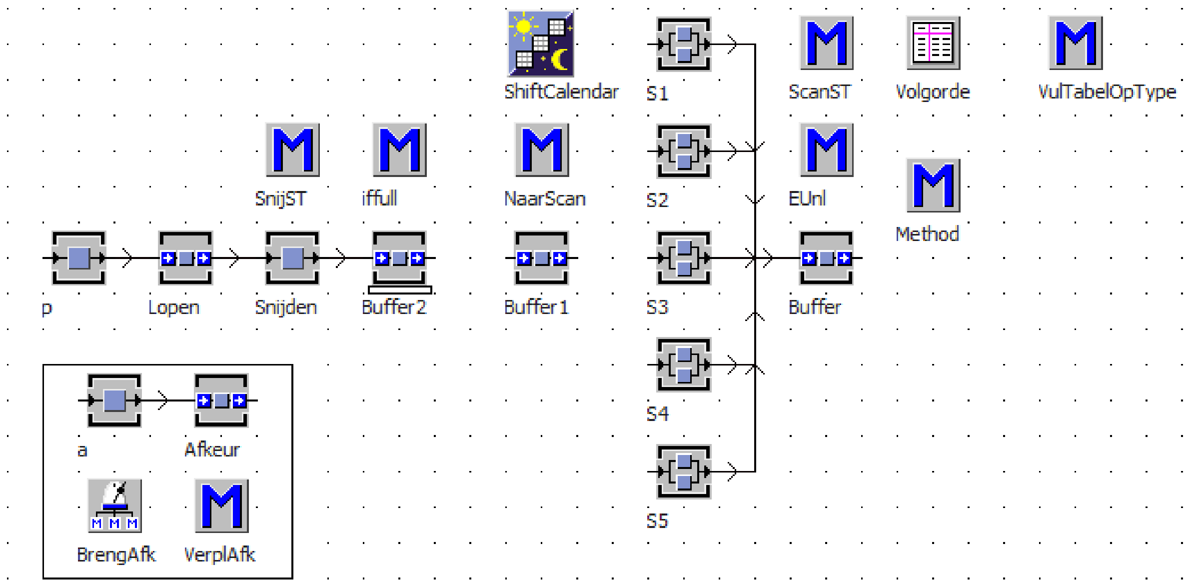


Figure A.4. Scanning process

When the products are registered they arrive at station p after which they are placed in the buffer (Lopen). When an employee is ready the products are cut to size on a FCFS basis. The first product that comes in the buffer (Lopen) will be the first to enter the cutting (Snijden) station when this station is free. S1 to S5 represent the 3D scanners. Currently Comfoor B.V. owns three 3D scanners. To be able to consider future expansion we make it possible to have up to five 3D scanners. On the main frame in the table settings for scanning (SettingsInscannen) the 3D scanners can be put on or off. In the table a fixed priority rule can be stated that the 3D scanners use. Is the first priority earmolds (OS) for the first scanning station (S1)? When the first scanning station is ready and there is an earmold in the buffer

(Buffer1) the first earmold in the table is moved to the first scanning station. The to scanner (NaarScan) method makes this move possible. This method is triggered as soon as a product enters the buffer (Buffer1) and first triggers the fill in table (VulTabelOpType) method which fills in the table order (Volgorde) with information on the entered product. After filling in the table the method lets the product wait until a working 3D scanner is available. When a 3D scanner becomes available the table logic is checked. The product with the highest priority is moved to the scanner. Just like previous module the calendar (ShiftCalendar) determines the working times of the cutting station and the 3D scanners which is in this case 7:00-16:00. When ready the product arrives at the buffer (Buffer) and triggers the method (Method). This method moves the product onto the next module and records the user-defined attribute time of scanning (Ingescant), which is the time the product leaves the 3D scanner. The units in the black square represent the error handling department (Product Administratie). Here the products are send when they receive an error status. The generator move errors (BrengAfk) gives the products that had an error and are stored in buffer errors (Afkeur) back to the scanning department at 13:00 with the method move errors (VerplAfk).

Model

In Figure A.5 we see the contents of the module model (Modelleren). After scanning a digital print of the earmold exists and this can be transformed into a digital print of the product itself.

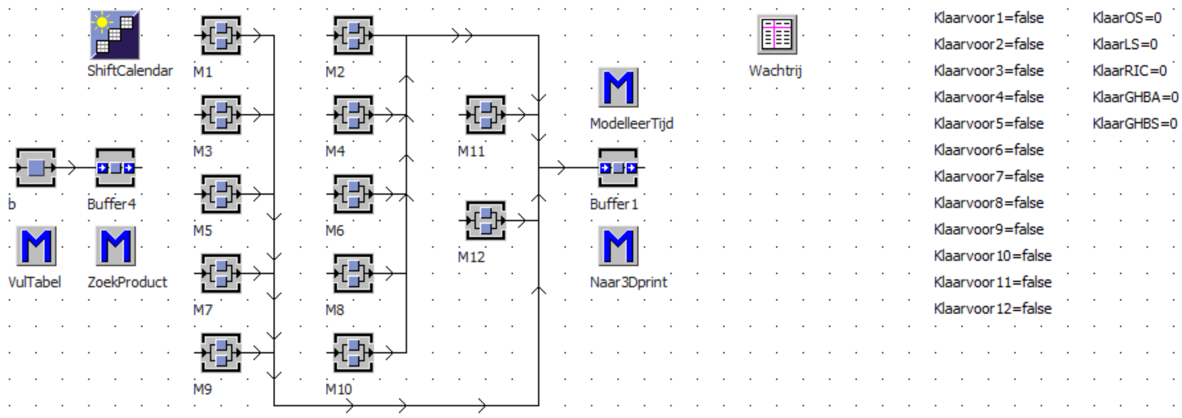


Figure A.5. Modelling process

When the products are scanned they arrive at the incoming station (b) and are moved to the buffer (Buffer4). Entering the buffer triggers the method wait until free (WachttotVrij) which first triggers the method fill in table (VulTabel) that fills up the table queue (Wachtrij) with the entered product's information and afterwards lets the product wait until one of the modelling stations (M1 to M12) is available. Comfoor B.V. currently owns 10 modelling stations of which 9 have the required licence to be able to work without limitations. Again, extra capacity is made available to be able to see the impact of adding extra workstations. These modelling stations can all have different priorities (products that will be handled first) which can be stated in the table settings for modelling (SettingsModelleren) on the main frame. There is also an option to turn workstations on and off in this same settings table. After a workstation has become available the method search for product (ZoekProduct) is activated. First this method checks which workstation has become available. When it has found one it checks from high to low priority if there is a product that the workstation can handle. If the product is found it is moved to the workstation. Before entering the workstation the method modelling time (Modeleertijd) sets the right processing time for the workstation which is received from the user-defined attribute modelling service time (ModelleerST) attached to the product. The calendar (ShiftCalendar) governs the working hours. There are two shifts, one from 6:00 – 12:00 and one from 13:00 to 19:00. When finished the products enter the buffer (Buffer1) and trigger the method to 3D printing (Naar3Dprint) which moves the products to the next module which is 3D printing. Additionally, it saves the time at which the product was finished at modelling in the user-defined attribute modelling done (Gemodelleert). The variables ready for 1 to ready for 12 (Klaarvoor1 until Klaarvoor12) keep track if there is a product in buffer (Buffer4) that can be produced on that station. Ready for 1 (Klaarvoor1) determines if there is a product that can be produced on work station M1, etc.. The variables ready earmolds (KlaarOS) etc. Keep track of how many products have been produced on a day. To later calculate the amount of time the modellers have spend on modelling in the table productivity on the main frame.

3D Printing

After a model of the product is made it can be put into the 3D printer. However, first enough models must be present to be able to make a print job that fills up one 3D printer.

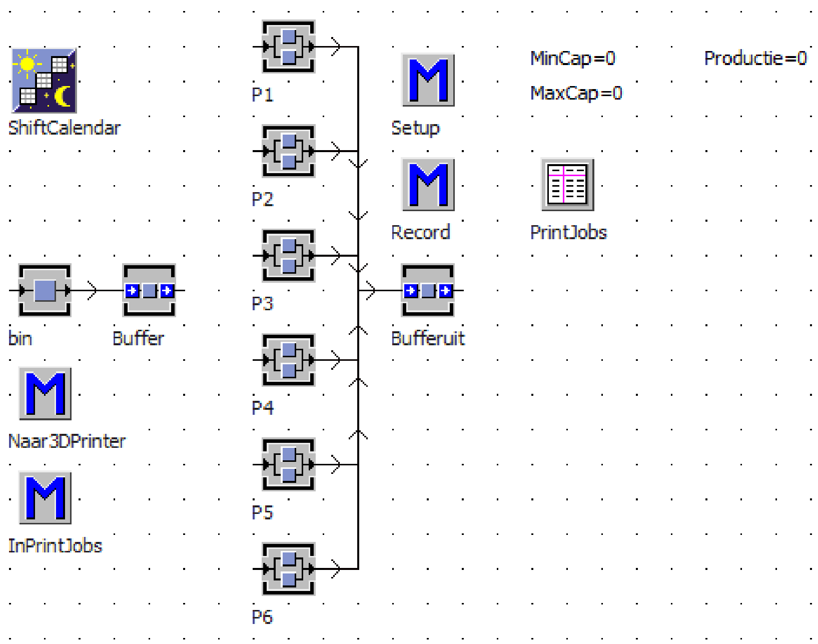


Figure A.6. 3D printing process.

When the products are finished at modelling they arrive at arriving station (bin) before arriving in the waiting area (Buffer). Entering the waiting area triggers the method to 3D printer (Naar3DPrinter) which looks if we can move products to the printer and moves them when possible. The variable minimum capacity (MinCap) represents the minimal capacity each machine needs, which comes down to the printer(s) that have the least capacity. Maximum capacity (MaxCap) is the opposite. It represents the capacity of the 3Dprinter(s) with the highest capacity. The to 3D printer method first triggers the go to print job (InPrintJobs) method which fills table print jobs (PrintJobs) with information on the entered product (its IdNo and the object itself). Afterwards it looks if there is a product type of which there are more or equal to the amount listed in the table settings for batch size on the main frame. Only after there are enough products available the method continues. It checks which 3D printer is available. Next it loops through the priorities of the printer and the first priority it stumbles upon that has enough products to fill the printer it moves as much products as indicated by the batch size that belongs to the chosen 3D printer and the product type. After a product is moved its information is deleted from the table print jobs and the method is terminated after the 3D printer is full. If no 3D printer can be filled with the number of products (for example the minimal capacity is 30, there are 35 products available, but the only 3D printer that is available has a capacity of 45) a suspended method is created which waits until there is at least the maximum capacity number of products available. When products arrive at the 3D printer the method setup (Setup) triggers. After the 3D printer is full it looks at the settings for the 3D printer (Settings3Dprinter) table at the amount of time it takes to take the last print job out and let the 3D printer start a new print job. It waits to process the products until this time has passed. When ready the products move to the exit buffer (Bufferuit) and trigger the method record (Record). This method submits the time the product finished at the 3D printer into the user-defined attribute 3D printing ready (Ge3Dprint). Afterwards it moves the product to the next module. The calendar (ShiftCalendar) is set from 6:00-20:00.

Cleaning products
In figure A.7 we see the contents of the cleaning (_3DprintSchoon) module. After a 3D printer is ready the fluid from which the products are made still lingers in the nooks and crannies of the product. This liquid needs to be removed before it gets solid as well and ruins the product.

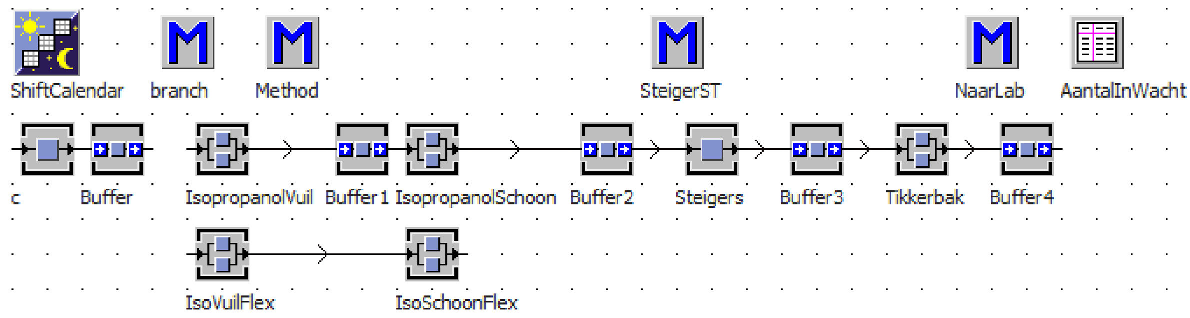


Figure A.7. Cleaning process.

When ready at 3D printing the products arrive at station c and continue to the waiting area (Buffer). When available products arrive at the dirty cleaning station (IsopropanolVuil) which triggers the method (Method). The method changes the processing time according to the user-defined attribute cleaning service time (ISOST). The cleaning stations dirty cleaning station (IsopropanolVuil) and clean cleaning station (IsopropanolSchoon) have the same processing times. Therefore, the same method also applies at this station. For silicone earplugs a different processing time counts, therefore the aforementioned stations also have a silicone variant (IsoVuilFlex and IsoSchoonFlex). Silicone earplugs are routed from the first buffer (Buffer) with a method (Branch) to the silicone cleaning station. From the second silicone cleaning station the silicone earplugs are routed to the second buffer (Buffer2) with the same method (Branch). As can be seen above in Figure A.7 the rest of the process follows in a linear fashion. When ready, the product has arrived in the buffer (Buffer4), the to lab (NaarLab) method is triggered which records the time the product is ready at the cleaning process in the user-defined attribute all clean (Schoongemaakt). To move to the next module, we first check the name of the product. Earmolds (OS) and acrylate earplugs are moved to the polishing machine (Trommel) module. The LifeShells (LS) and receiver in ear's (RIC) are moved to the finishing (Afwerken) module. Lastly the silicone earplugs are moved to the preparing silicone earplugs (InspuitenPellen) module. The calendar (ShiftCalendar) sets the time the stations work the same as the 3D printers, from 6:00 to 20:00.

Polishing machine

The earmolds (OS) and the acrylate earplugs need to be put in the polishing machine (Trommel) to make the product smoother.

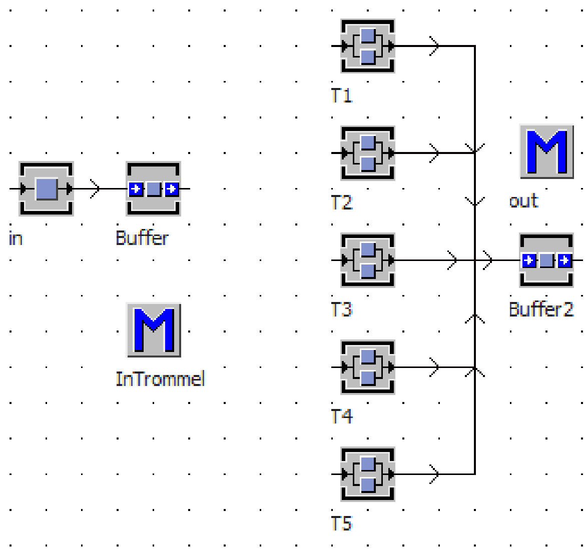


Figure A.8. Polishing machine process.

Earmolds (OS) and acrylate earplugs arrive in the incoming station (in) and comes into the waiting area (Buffer). When entering the waiting area, the method in polishing machine (InTrommel) is triggered. This method sends the products to the first polishing machine that is available. When a polishing machine is available but already is busy processing products of a different type the product is moved to the next available polishing machine that processes products of the same type or is empty. When moved, the product immediately sets the processing time of the polishing machine to the correct value with its user-defined attribute service time polishing machine (TrommelST). The calendar (ShiftCalendar) dictates working hours, which are from 7:00-16:00. When products are ready they go to to the second buffer (Buffer2) which triggers the method out (out). out stores information on what time the product is ready at the polishing machines in the user-defined attribute ready with polishing (Getrommelt). Afterwards it sends the products to the next module which is finishing (Afwerken).

Preparing silicone earplugs

After 3D printing the silicone earplugs product is nothing more than an empty shell. The silicone material the real product consists of first needs to be injected into the cast print. When this material has become solid, the cast print can be peeled off and the product is ready for finishing.

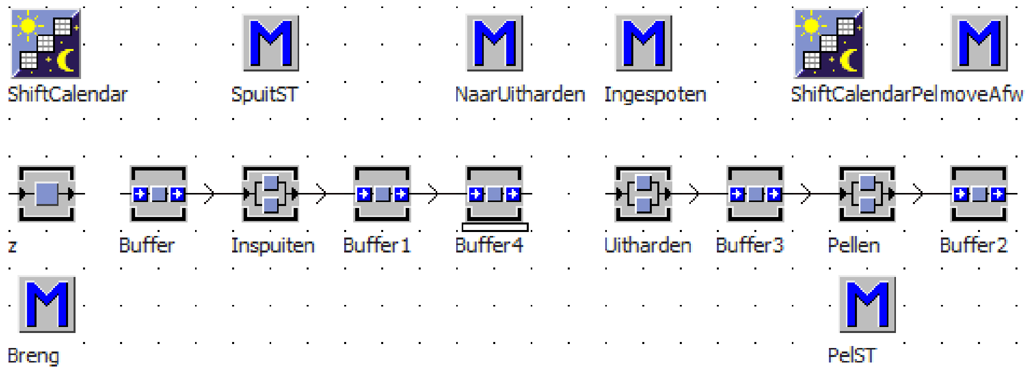


Figure A.9. Injection and peeling process.

After being cleaned the silicone earplugs products arrive at the incoming station (z) and go further into the waiting area (Buffer). When the injection station (Inspuiten) is available the products are processed and continue where they eventually arrive at the third buffer (Buffer4). Here the products are batched to go to the ovens (Uitharden). The batch size can be determined in the table setting the batch size (SettingsBatchSize) on the main frame. So when a product enters the third buffer it triggers the to the ovens method (NaarUitharden) which waits until the third buffer is full and then moves all products to the ovens when there is space available. After the products leave the ovens and enter the fourth buffer (Buffer3) the method injected (Ingespoten) records the time the products were ready with the injection process in the user-defined attribute ready with injection (Ingespoten). The product moves along to peeling off cast (Pellen) and leaves the module once entering the fifth buffer (Buffer2) through the move to finishing method (moveAfw). The move to finishing method stores the time the product has been relieved of its cast print in the user-defined attribute peeling done (Gepelt). Furthermore, it moves the product to the next module. Again, the calendar (ShiftCalendar) is used to set working hours, which are between 7:00 and 16:00.

Finishing

When products are ready to be finished they arrive at the finishing module (Afwerken).

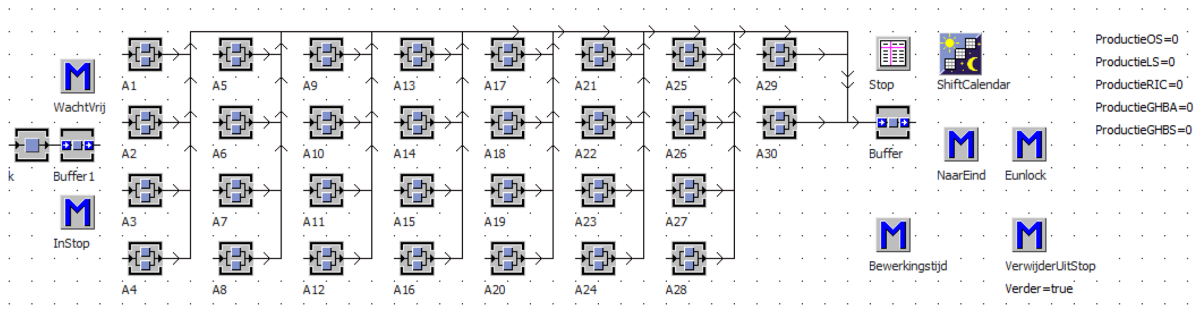


Figure A.10. Finishing process.

Products arrive in the incoming station (k) and go forth to the waiting area (Buffer1). There they trigger the method wait until free (WachtVrij) which first triggers the method put on hold (InStop). The put-on hold method registers arriving products into the table stop (Stop). Next the wait until free method checks if there are enough products of the same type in the buffer. This has to be equal or more than the batch size that is set in the table settings of batch size (SettingsBatchSize) on the main frame. When there are enough products of the same type, the method continues only if there is an employee (stations A1, …, A30) available, otherwise it repeats the method after 1 hour. These employees can be "turned on and off" in the table settings for finishing (SettingsAfwerken), also their priorities or types of products they can process are listed there and can be changed. When an employee is free, it loops through all employees and checks which one is available. When it finds an available employee, it loops through the priorities and compares priority types with the number of available products per type. The first priority type that has enough products waiting in the buffer will be picked for moving the required number of products to the employee. After the products are moved they are deleted from the table stop and the method terminates. When no priority types of the employee are available the method terminates and is restarted after 1 hour. When there are no product types that have equal or more products than the required batch size the method terminates. When products are moved to the employee the method production time (Bewerkingstijd) triggers. It sets the processing time of the employee to the number of products that need to be processed times the time it takes to process one product. This processing time is listed in the user-defined attribute finishing processing time (AfwerkST). When finished the products arrive in the buffer (Buffer). This triggers the method to final check (NaarEind), which lists the time the products are finished in the user-defined attribute finished (Afgewerkt) and moves them to the next module. The calendar (ShiftCalendar) sets the working hours between 7:00 and 16:00.

Final check

When a product is finished it needs to be checked for errors. This is simulated in the final check (Eindcontrole) module.

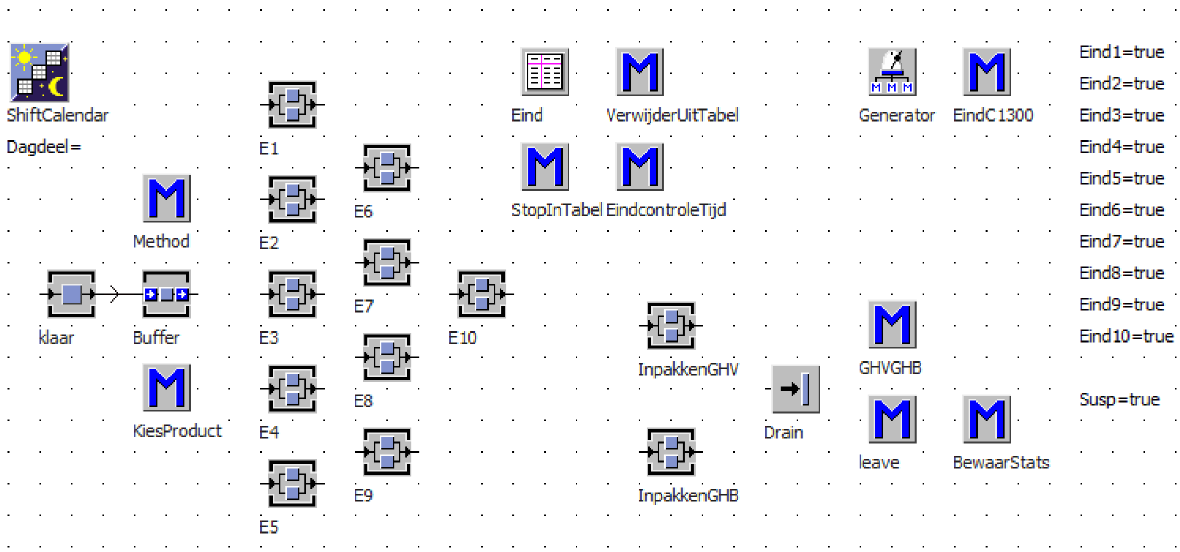


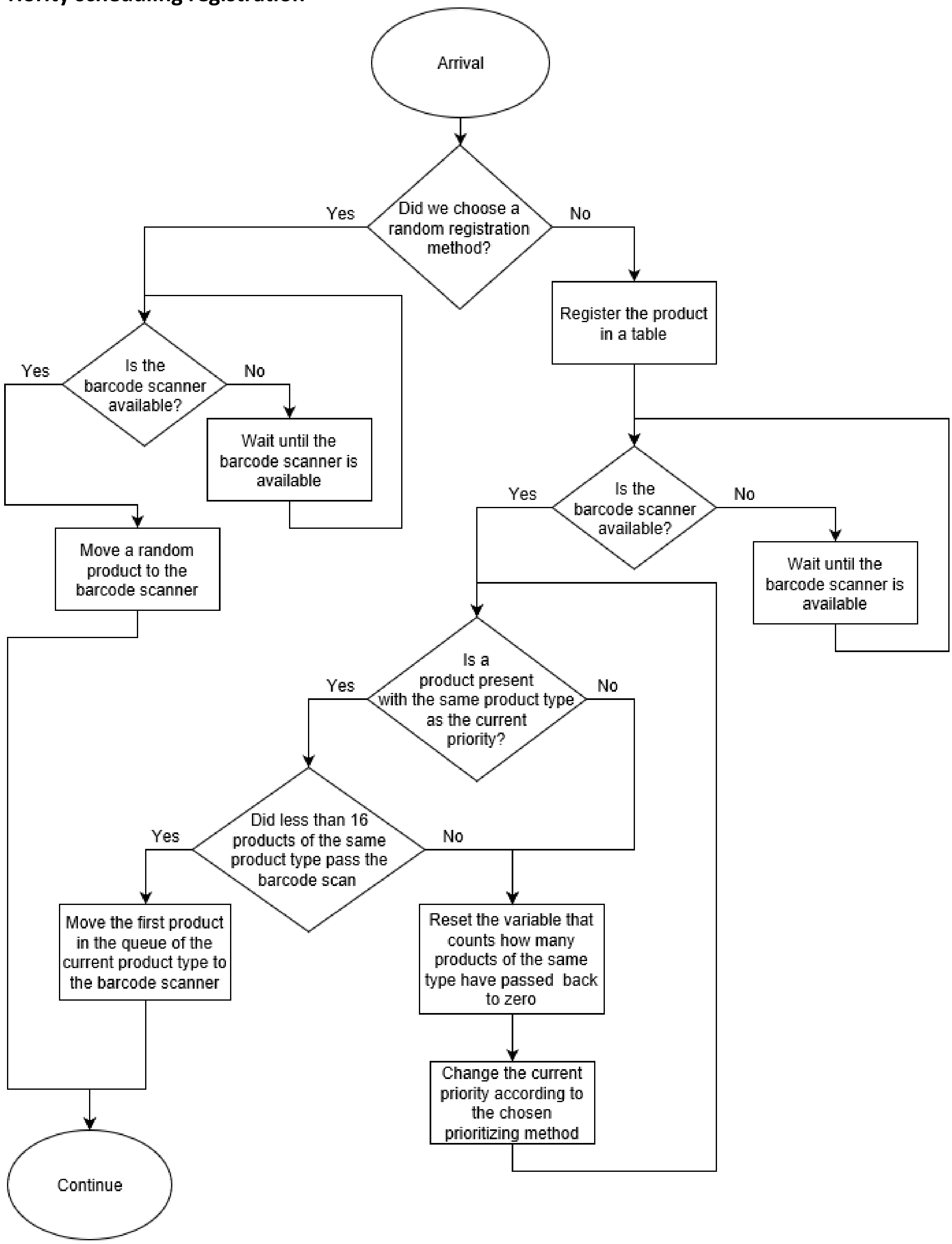
Figure A.11. Final check process.

Products arrive at incoming station ready (klaar) and move to the buffer (Buffer). Method (Method) is triggered. This works the same at the previous modules. First the products are registered in the table final (Eind) with the method put it in the table (StopInTabel). Afterwards the method is repeated every minute until an employee (E1 until E10) becomes available. When available the method, choose a product (KiesProduct), is triggered. First the variables final 1 (Eind1) to final 10 (Eind10) are updated. These variables keep track if there are currently products in the buffer that the employee has in his/her priority list. These priorities and the number of employees can be changed in the table settings for final check (SettingsEindC). Next the method checks which employee is available. The first priority of which there are products in the buffer will be moved. When the product is moved the remove from table (VerwijderUitTabel) method is triggered. This method looks at all products that are currently processed by employees and when these products are still listed in the table it removes them from the table. When product information is deleted from the table the suspend (Susp) variable is set to false. This variable is added to counter unwanted code behaviour due to the same method repeating itself and interrupting each other to move already moved products before they are deleted. With the suspend variable this is not possible anymore. Now when suspend is true the method continues, when false the method is terminated. When repeated the method choose product (KiesProduct) starts from the line "while bool = false loop". When a product enters a station of an employee it sets the production time of the station to the value stated in the table settings for production time (SettingsProductieTijd) which is stored into the user-defined attribute final check service time (EindcontroleST). When ready the product triggers the method earmolds and earplugs (GHVGHB) which checks the product type of the product. When it is earmolds it goes to the station send-off earmold (InpakkenGHV). Before moving the simulation time is recorded on when the product is ready at final check in the user-defined attribute ready (Goedgekeurt). When the product is of the category earplugs it moves to send-off earplug (InpakkenGHB). When finished the product moves to the drain. When it enters the drain the method keep statistics (BewaarStats) is triggered. This method registers all the times the product has left the production stages into the table Statistics (Statistieken) on the main frame. Next it calculates the total time the product has resided in the system plus if this time is within the predetermined limits. These limits can be set on the main frame in the table settings production time (SettingsLeadTijd). Lastly there is the generator (Generator) and the method final check at 13:00 o clock (EindC1300). The generator creates an event which triggers the method final check at 13:00 every day at 13:00. The method changes the number of employees present at final check to its afternoon value and also changes the priorities of the employees. This to represent that the whole day someone is present to check earplugs, while in the afternoon the earmolds is checked. An additional job of this method is to update the variables of final 1 to final 10. This is needed because when the end of a simulation period has ended, and no other products are produced the products have to wait in the morning and no new products come in in the afternoon to change the values, leaving them on false and letting repeating methods, that repeat until they have had effect, repeat indefinitely. The calendar (ShiftCalendar) set the working hours from 7:00-16:00. The variable part of the day (Dagdeel) keeps track of the time of day. When it is morning the value is morning (Ochtend) and when after 13:00 the value is afternoon (Middag). The methods choose a product and final check at 14:00 use the variable part of the day to determine which rows they need in the table settings for final check (SettinsgEindC) to know the priorities and the number of available employees.

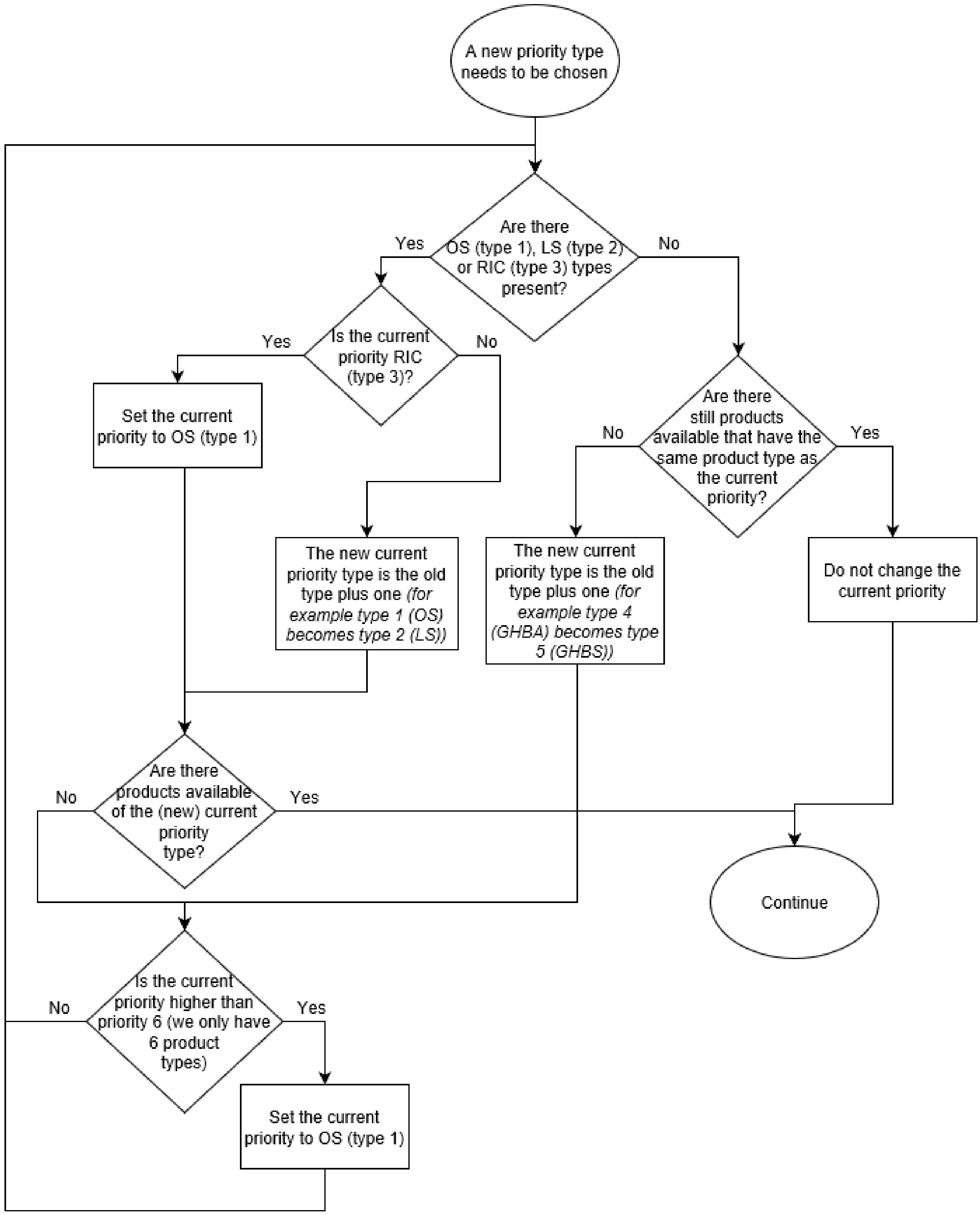
Appendix B – Flow Charts of the Key Processes

In this chapter the logic on how priority is given within the model is elaborated on by means of flow charts. Within the registration logic presented in the next sub chapter a new priority is chosen. This Priority depends on the selected method, which can be blind, current, flex first or equal priority. Of these priorities selecting methods blind picking is the only one without a flow chart because of its simple nature.

Priority scheduling registration

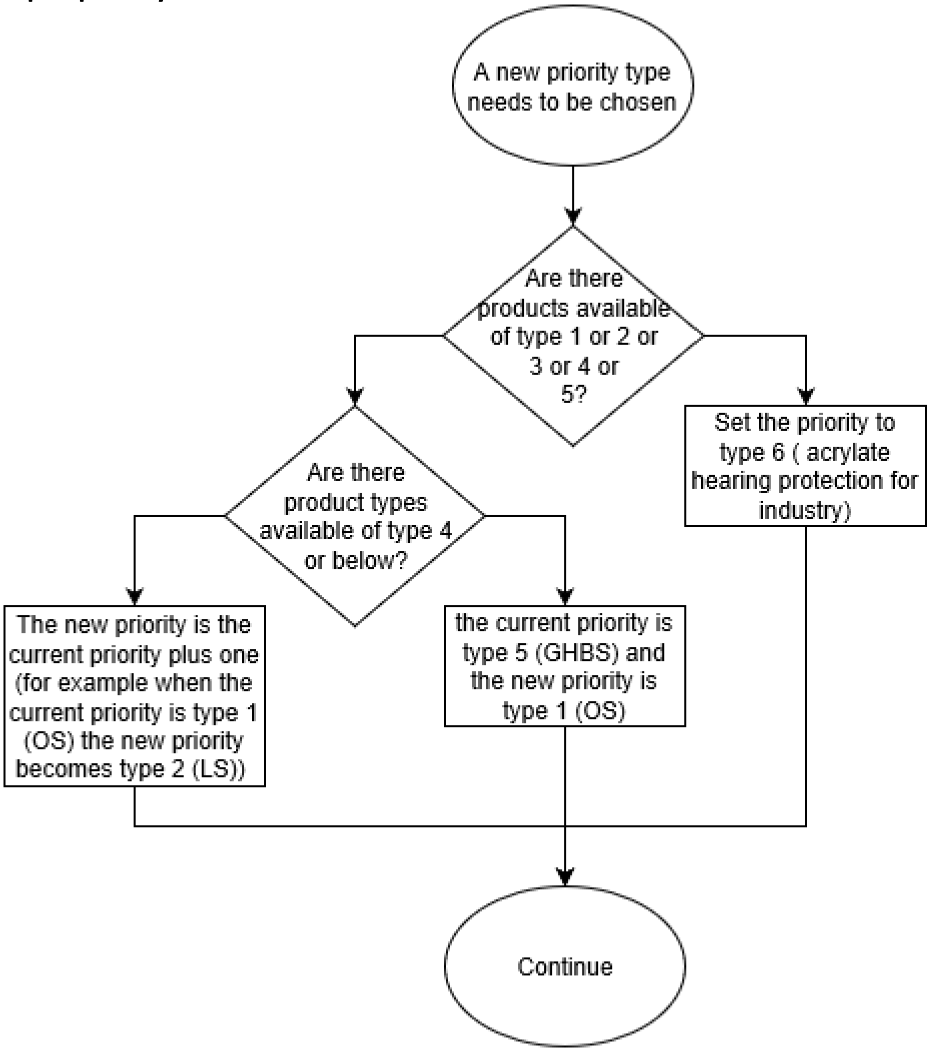


Current

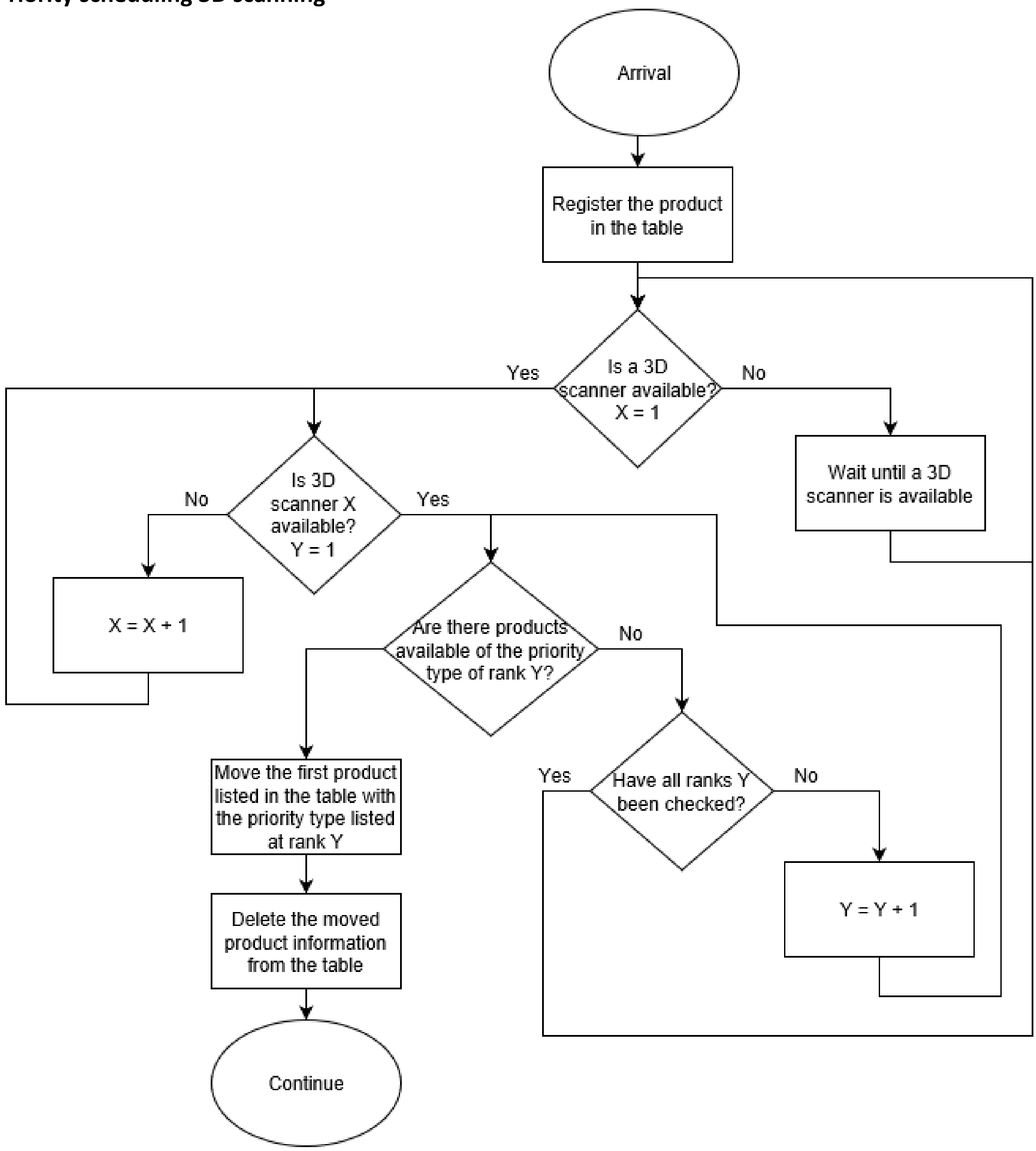


A new priority type needs to be chosen
Are there OS (type 1), LS (type 2) or RIC (type 3) types present?
Yes
No
Is the current priority RIC (type 3)?
Yes
No
Set the current priority to OS (type 1)
The new current priority type is the old type plus one *(for example type 1 (OS) becomes type 2 (LS))*
Are there still products available that have the same product type as the current priority?
No
Yes
The new current priority type is the old type plus one *(for example type 4 (GHBA) becomes type 5 (GHBS))*
Do not change the current priority
Are there products available of the (new) current priority type?
No
Yes
Continue
Is the current priority higher than priority 6 (we only have 6 product types)
No
Yes
Set the current priority to OS (type 1)

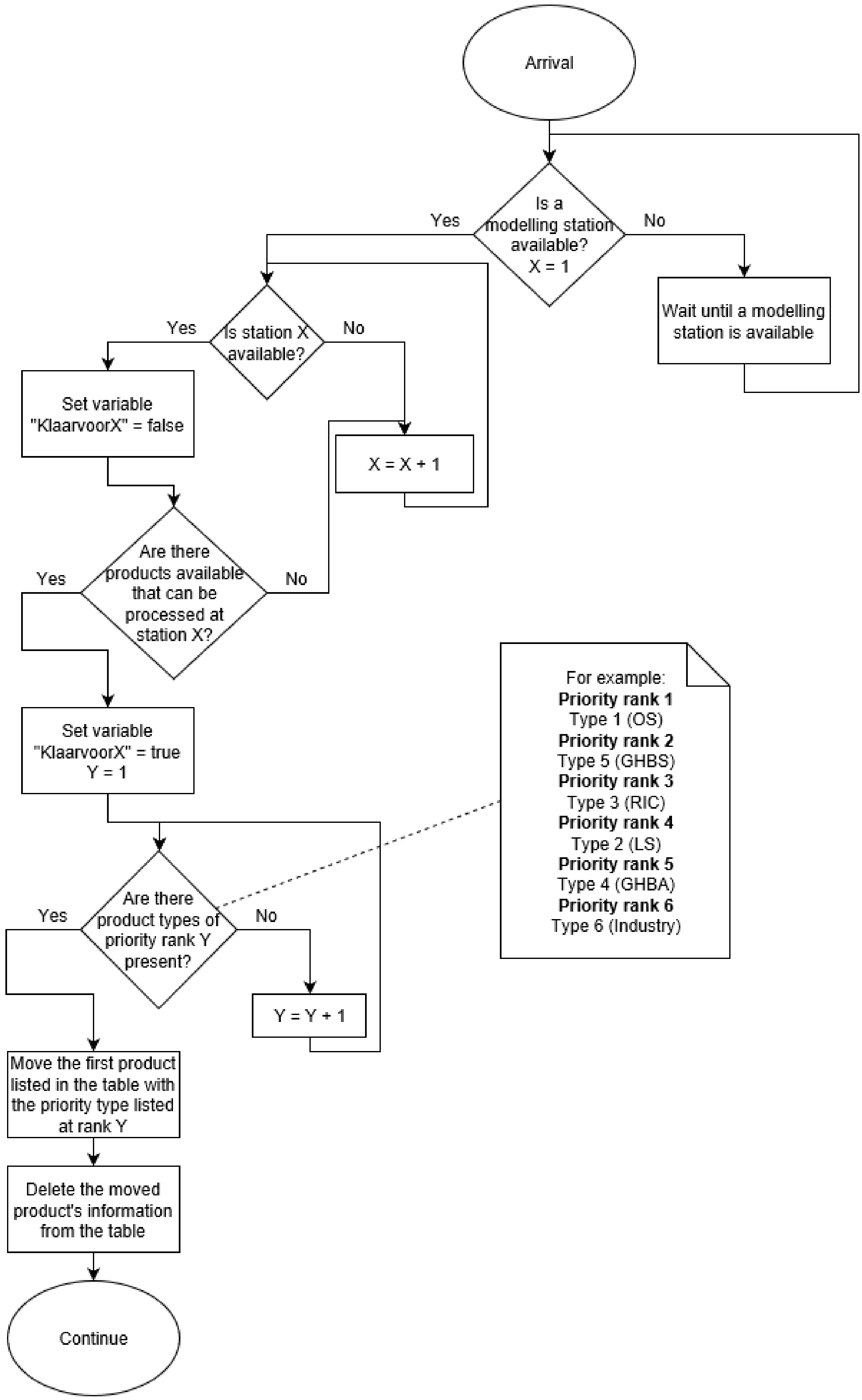
Equal priority



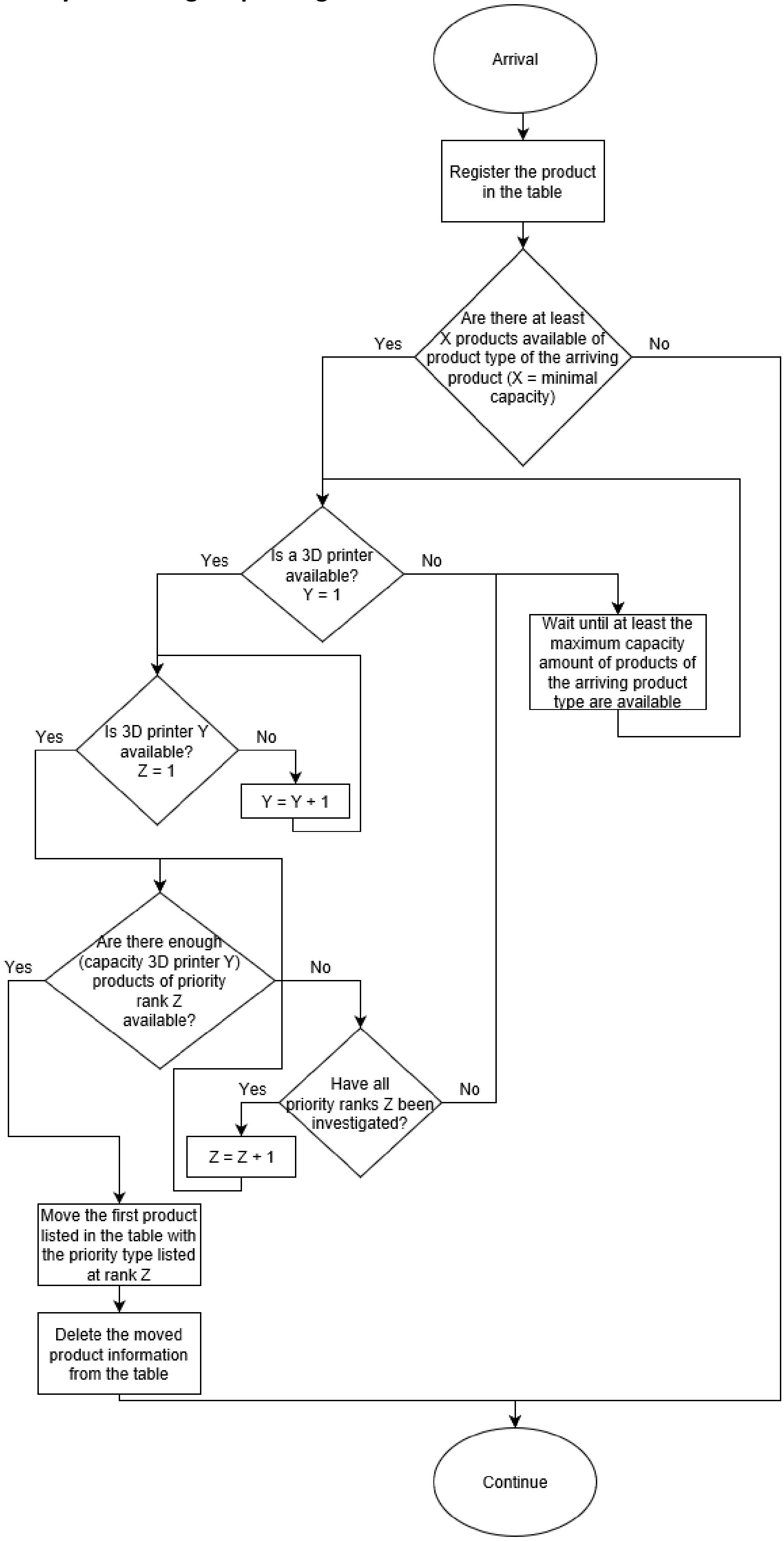
Priority scheduling 3D scanning



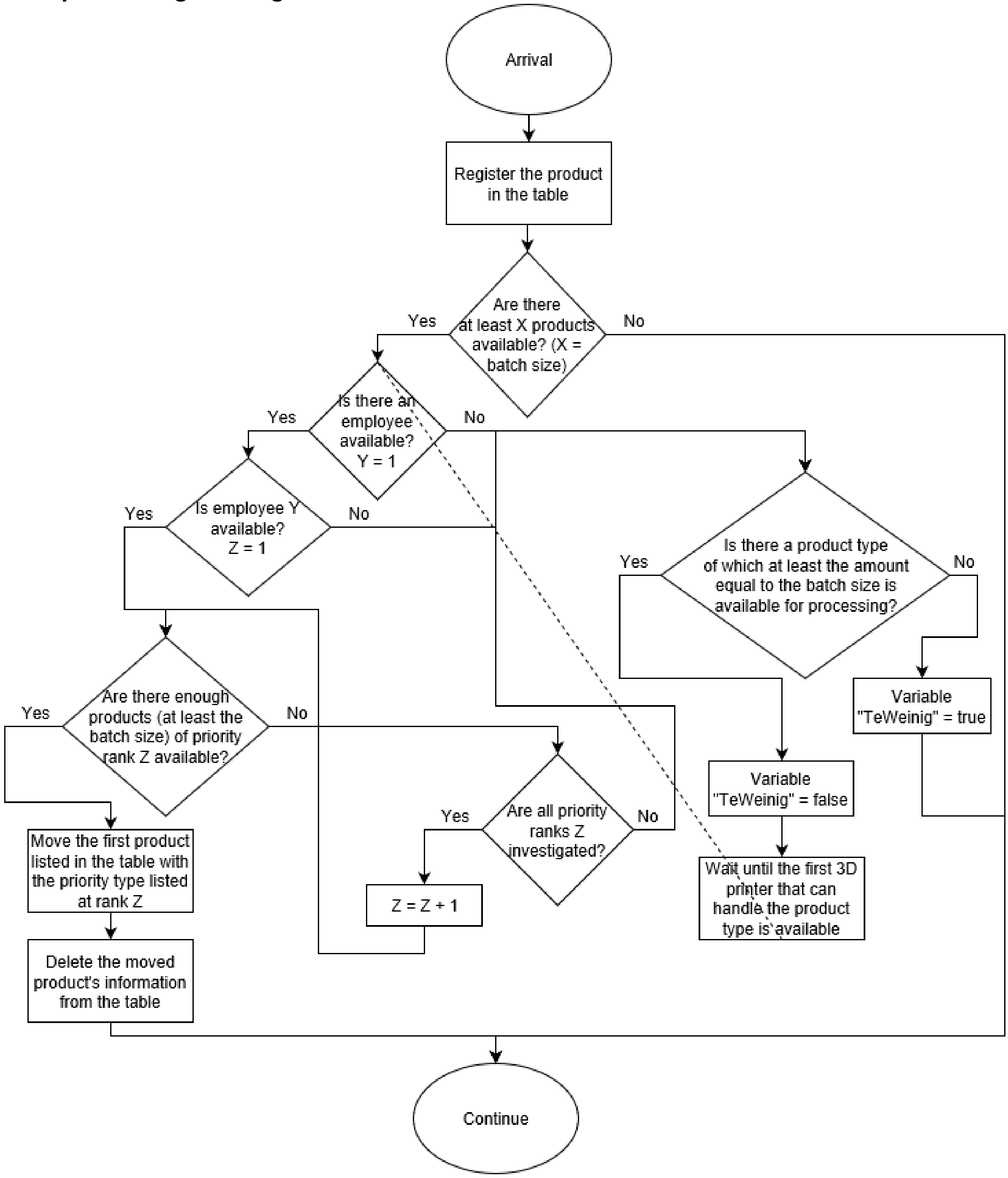
Priority scheduling modelling



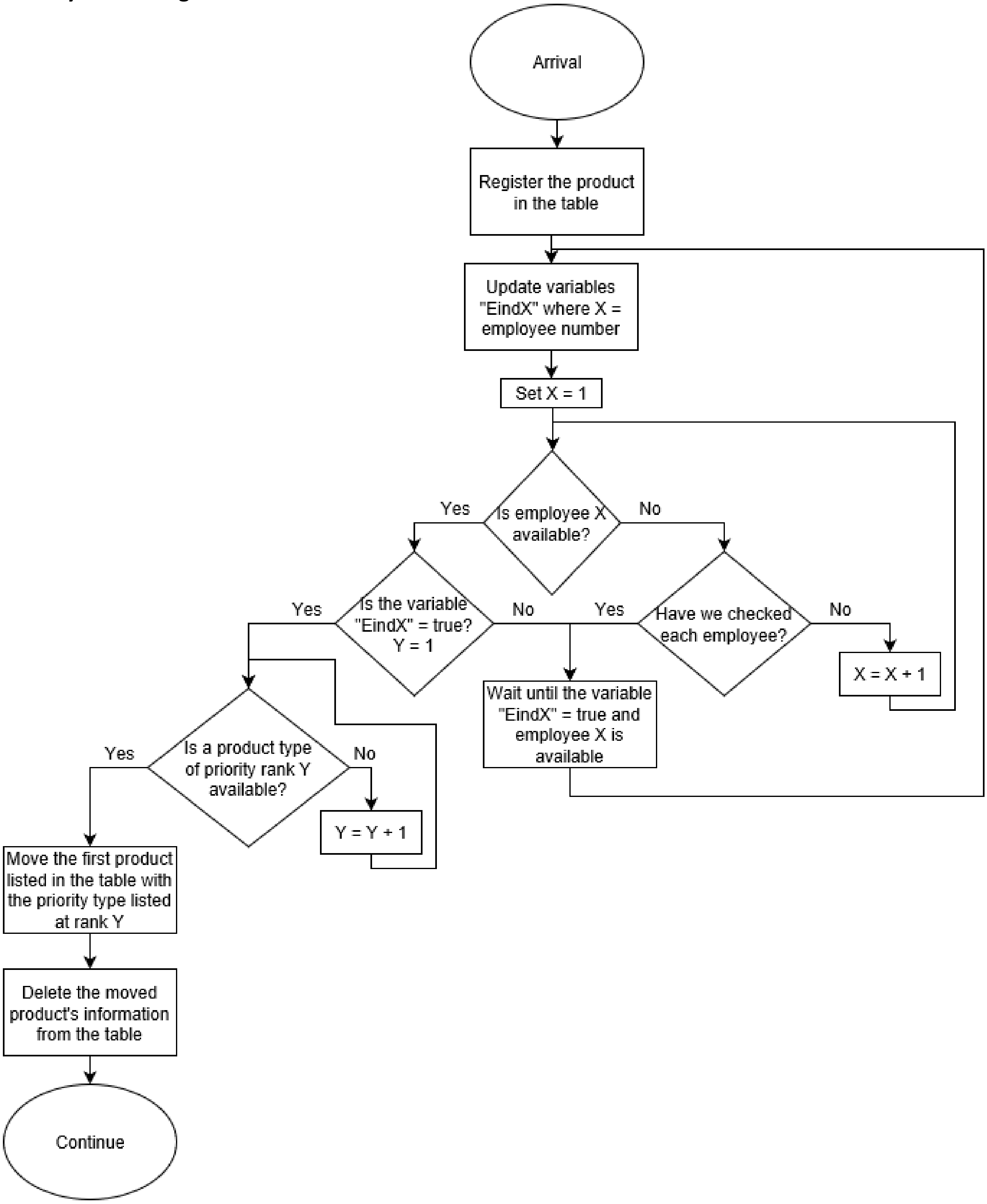
Priority scheduling 3D printing



Priority scheduling Finishing



Priority scheduling final check



Appendix C – PERT settings demand analysis 2017

For all eight scenarios, from January combined with February until December, we calculated the averages over the period covered by the scenario based on historical data of 2017. This data is total demand per product type that entered the system on that day. So, the first rows of the tables portray the average demand per day over the scenario period. Next, we determine the amount of employees needed to be dedicated to finishing to cover the daily demand of that product type. These needed employees are rounded up. Lastly, we see the amount of 3D printers necessary to keep a constant flow of products. All these resource requirements do not take into account that resources can be shared amongst the production of different product types. Therefore, these settings need to be refined using the simulation model. The averages cannot yet be defined through historical data for the months that have yet to come. Therefore, an estimate is made based on the demand of 2016 and the trend we see currently. In the row +/- 2016 we show the percentage we added or substracted to the data of 2016 to come up with the averages of 2017.

Jan-Feb	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	245	95	125	210	235	910
Finishers	5	3	3	3	5	19
Modellers	3	1	2	1	3	10
3D printers	2	1	1	1	2	7

Table C.1. PERT settings scenario Jan-Feb

Mar-Apr	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	245	95	120	240	235	935
Finishers	5	3	2	4	5	19
Modellers	3	1	1	2	3	10
3D printers	2	1	1	1	2	7

Table C.2. PERT settings scenario Mar-Apr

May	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	225	80	120	240	235	900
Finishers	4	2	2	4	5	17
Modellers	2	1	1	2	3	9
3D printers	2	1	1	1	2	7

Table C.3. PERT settings scenario May

<u>**June**</u>	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	225	80	110	120	390	925
Finishers	4	2	2	2	7	17
Modellers	2	1	1	1	4	9
3D printers	2	1	1	1	3	8

Table C.4. PERT settings scenario June

<u>**July**</u>	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	225	80	110	120	290	825
Finishers	4	2	2	2	5	15
Modellers	2	1	1	1	3	8
3D printers	2	1	1	1	2	7
+/- 2016	-11%	-24%	+14%	+31%	+33%	

Table C.5. PERT settings scenario July

<u>**August**</u>	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	180	65	100	120	235	700
Finishers	4	2	2	2	5	15
Modellers	2	1	1	1	3	8
3D printers	2	1	1	1	2	7
+/- 2016	-15%	-25%	+15%	+15%	+15%	

Table C.6. PERT settings scenario August

Sept-Nov	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	255	100	150	255	220	980
Finishers	5	3	3	4	4	19
Modellers	2	1	2	2	2	9
3D printers	2	1	1	1	2	7
+/- 2016	-10%	-20%	+10%	+15%	+15%	

Table C.7. PERT settings scenario Sept-Nov

December	**OS**	**LS**	**RIC**	**GHBA**	**GHBS**	**Total**
Average	225	85	145	255	185	895
Finishers	4	2	3	4	4	17
Modellers	2	1	2	2	2	9
3D printers	2	1	1	1	2	7
+/- 2016	-15%	-25%	+15%	+15%	+15%	

Table C.8. PERT settings scenario December

Appendix D – Experiments

Experimental design for the first round of experiments. We use the settings for the amount of finishers from Appendix C. However, we know that this amount is enough or more than enough to fulfil daily demand. Therefore, we see if we could do with one employee less. For example, we need for scenario Jan-Feb five OS finishers. In our experiments when the row underneath OS in table D.1 says true than OS has five finishers. If the row says false OS has four finishers. However, when we would find in Appendix C that we need two finishers. True means we use two and false means we use one.

#	OS	LS	RIC	GHBA	GHBS
1	TRUE	FALSE	FALSE	FALSE	FALSE
2	FALSE	TRUE	FALSE	FALSE	FALSE
3	FALSE	FALSE	TRUE	FALSE	FALSE
4	FALSE	FALSE	FALSE	TRUE	FALSE
5	FALSE	FALSE	FALSE	FALSE	TRUE
6	TRUE	TRUE	FALSE	FALSE	FALSE
7	TRUE	FALSE	TRUE	FALSE	FALSE
8	TRUE	FALSE	FALSE	TRUE	FALSE
9	TRUE	FALSE	FALSE	FALSE	TRUE
10	FALSE	TRUE	TRUE	FALSE	FALSE
11	FALSE	TRUE	FALSE	TRUE	FALSE
12	FALSE	TRUE	FALSE	FALSE	TRUE
13	FALSE	FALSE	TRUE	TRUE	FALSE
14	FALSE	FALSE	TRUE	FALSE	TRUE
15	FALSE	FALSE	FALSE	TRUE	TRUE
16	TRUE	TRUE	TRUE	FALSE	FALSE
17	TRUE	TRUE	FALSE	TRUE	FALSE
18	TRUE	TRUE	FALSE	FALSE	TRUE
19	TRUE	FALSE	TRUE	TRUE	FALSE
20	TRUE	FALSE	FALSE	TRUE	TRUE

21	TRUE	FALSE	TRUE	FALSE	TRUE
22	FALSE	TRUE	TRUE	TRUE	FALSE
23	FALSE	TRUE	FALSE	TRUE	TRUE
24	FALSE	TRUE	TRUE	FALSE	TRUE
25	FALSE	FALSE	TRUE	TRUE	TRUE
26	FALSE	TRUE	TRUE	TRUE	TRUE
27	TRUE	FALSE	TRUE	TRUE	TRUE
28	TRUE	TRUE	FALSE	TRUE	TRUE
29	TRUE	TRUE	TRUE	FALSE	TRUE
30	TRUE	TRUE	TRUE	TRUE	FALSE
31	TRUE	TRUE	TRUE	TRUE	TRUE
32	FALSE	FALSE	FALSE	FALSE	FALSE

Table D.1. experimental design first set of simulation experiments