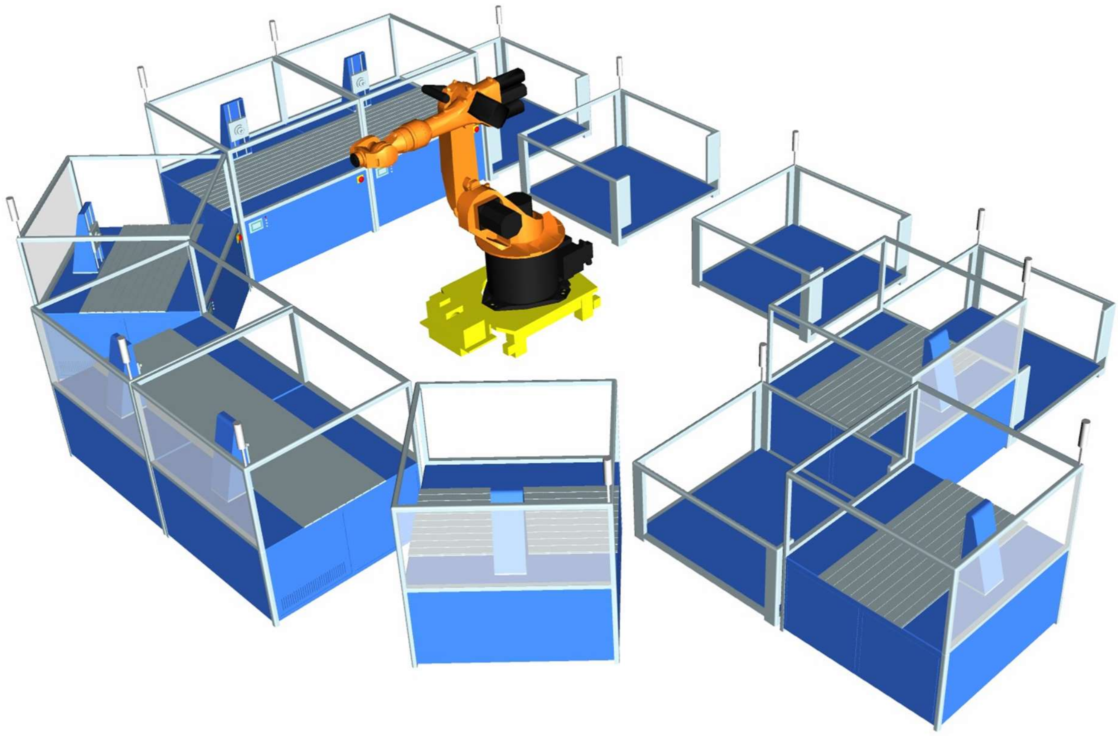
CHALMERS
UNIVERSITY OF TECHNOLOGY



# Viability of Discrete Event Simulation in the Early Design of Production Systems

## A Case Study in the Design of a Flexible Automation Cell

Master's thesis in Production Engineering


Michael Bekk
Erik Moen

DEPARTMENT OF INDUSTRIAL & MATERIALS SCIENCE



CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2020
www.chalmers.se

Master's thesis 2020

# Viability of Discrete Event Simulation in the Early Design of Production Systems

A Case Study in the Design of a Flexible Automation Cell

MICHAEL BEKK
ERIK MOEN

Department of Industrial and Materials Science
*Division of Production Systems*
Chalmers University of Technology
Gothenburg, Sweden 2020

Viability of Discrete Event Simulation in the Early Design of Production Systems
A Case Study in the Design of a Flexible Automation Cell
MICHAEL BEKK, ERIK MOEN



Supervisor: Camilla Lundberg, Department of Industrial and Materials Science
Supervisor: Ana Esther Bonilla Hernandez, GKN Aerospace Engine Systems
Examiner: Anders Skoogh, Department of Industrial and Materials Science

Master's Thesis 2020
Department of Industrial and Materials Science
Division of Production Systems
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: 3D-model of the flexible automation cell built using Tecnomatix Plant Simulation.

Typeset in LaTeX
Gothenburg, Sweden 2020

Viability of Discrete Event Simulation in the Early Design of Production Systems
A Case Study in the Design of a Flexible Automation Cell
MICHAEL BEKK
ERIK MOEN

# Abstract

In the manufacturing industry, Discrete Event Simulation (DES) is recognised as a tool utilised in the analysis and optimisation of production systems. The effectiveness of DES, however, depends mainly on the data available from the system which is to be simulated and the quality of this data. This thesis investigates the viability of DES when a production system is in its early design stages with limited data available, and no complete system of which to translate into a virtual model. A case study was conducted where a production system in its early design stages was modelled and analysed by applying DES. Following the case study, the results of the case were analysed and a consensus was formed whether DES was usable as a tool to assist the system developers in the development process. Despite a lack of high-quality data, the case produced useful results for the system developers, and a clear trend of the performance and behaviour of the system was observed. Improvement suggestions were passed on to the system developers to assist the developers and as an extension save time and money. With the support of successful case results, substantial proof that DES is viable to use as a design tool in the early design of production systems were found. However, this is not without its challenges, as the majority of the required data and the behaviour of the system has to be estimated, limiting the accuracy of the results. Despite these challenges, DES is a viable approach, and can be utilised to influence design changes and parameter improvements of production systems in their early design stages.

Keywords: discrete event simulation, discrete event systems, case-study, production systems, early design, viability, simulation.

# Acknowledgements

Firstly, we would like to extend our sincere gratitude to our supervisor at Chalmers, Camilla Lundberg. Her continued support and interest throughout the entire process of the thesis was paramount in the success of this thesis. Secondly, we would like to thank our supervisor at GKN Aerospace Engine Systems, Ana Esther Bonilla Hernandez. Her guidance and continuous assistance and feedback has been much appreciated, and with her unending commitment and interest made this thesis a possibility. Furthermore, we thank GKN Aerospace Engine Systems for their welcoming environment and cheerful personnel. We would also like to thank Johan Nordling at Siemens Digital Industries Software and the selfless people at the Plant Simulation forums for their helpful tips and guidance in Tecnomatix Plant Simulation. Lastly, we would like to thank our examiner Anders Skoogh for examining this thesis.

Michael Bekk & Erik Moen, Gothenburg, May

# Contents

# Glossary

**Automated Guided Vehicle (AGV)**
A automatic truck or vehicle used within production to transport material and products.

**Collet**
A type of clamp which holds the work tool in place.

**Continuous simulation**
Variant of simulation commonly used when modelling a dynamic system where time flows continuously.

**Data**
Information often in the form of numbers which is gathered and used in many different applications.

**Discrete Event Simulation (DES)**
Variant of simulation which looks at discrete steps in time where an event represents a state change of the system.

**Sequential function chart (SFC)**
A chart showing the behaviour of a system sequentially. It includes states, operations and conditions of the system.

**Probability density function (PDF)**
A function depending on a random variable $X$ where the integral of the function gives the likelihood that the value of $X$ lies in the same interval as the integral.

**Production Cell**
A enclosed area within a production system containing a robot, a controller and other tools required to produce components or parts of components.

**Production System**
A system consisting of inputs, processes and outputs. These processes can be assembly, drilling etc.

**Rapid Entire Body Assessment (REBA)**
Ergonomic tool to assess the strain on the entire body from a certain pose.

**Rapid Upper Limb Assessment (RULA)**
Ergonomic tool to assess the strain on the body of a certain pose, mostly focused on the strain to the upper body.

**RobotStudio**
Robot programming and simulation software created by ABB Robotics.

**Tecnomatix Plant Simulation**
Graphical DES software developed by the company Tecnomatix.

**Theory of Constraints (TOC)**
A bottleneck breaking method developed by Eliyahu M. Goldratt [1] where one finds the bottleneck, tries different ways to break it and iterate until the bottleneck either has shifted or has been broken.

# List of Figures

# List of Tables

# 1
# Introduction

*This chapter introduces the project and its background. Followed by this, the aim, purpose, limitations and research questions of the project will be presented.*

## 1.1 Background

GKN Aerospace Engine Systems, from here on referred as GKN, produces aircraft engines components for aircraft engines and rocket components [2]. In the recent couple of years GKN has increased its production volumes of specific components noticeably, therefore the company is looking to automatise parts of its production to increase its productivity further. The produced components require high quality materials, common in aerospace manufacturing [3]. However, these materials usually have very low machinability, and as such, the tools that are used to machine the materials are worn out in a short amount of time [4]. The effect of this is that a substantial amount of tools have to be replaced everyday.

To machine the materials, GKN utilises a special kind of CNC-machine within its production. The tool replacement procedure of the machine consists of the following steps performed by an operator:

1. The machine cutting tool magazine is emptied of used cutting tools
2. New cutting tools are prepared
3. The new cutting tools are placed within the magazine

The process of preparing the tools is tedious and time-consuming, and in addition to the huge amount of time being spent on these tasks, there are also ergonomic issues related to the handling of the cutting tools, e.g. lifting, fastening and placing.

To solve this problem, GKN is currently researching and developing a flexible automation cell that might be capable of preparing the cutting tools that are used in the CNC-machines. The concept of the flexible automation cell is that it will have multiple modules which all have unique features. The modules can be changed depending on what type of work the cell should perform, thus making it flexible. This flexible automation cell could prepare the necessary tools for the machines faster and more efficiently than the current solution. This would save time, be more cost efficient, provide higher quality and reduce the physical strain that the operators are exposed to today.

As the flexible automation cell is still a work-in-progress, the overall design of the automation cell is not finalised. A method to assist the researchers in making design choices, flow balancing and bottleneck detection is to create a discrete event simulation (DES) model. A DES model can be a cost effective and time efficient method when designing new factory layouts as well as finding improvements to current layouts [5]. Rather than making changes and trying different solutions in the physical environment, one can make a virtual model with which to experiment without physically making changes in the real-life factory. Without having to stop production in order to experiment with improvements, one can save money and time [5], [6].

## 1.2 Purpose and aim

Finding design errors and improvement potentials early in the development phase of production systems has the possibility to save both time and money for manufacturing companies, and DES might be able to find these errors and potential improvements. The project may also provide important insight into a DES project in the early design of production systems and its potential and challenges.

Therefore, the aim of this project is to investigate whether DES is a suitable tool in the design stages of a completely new kind of production system where data is scarce. To investigate this, the aim is to model the flexible automation cell and create a DES of the cell within a production system, using this model as a test case. The simulation will be used to analyse the performance of the cell to balance and improve the sequencing of the tasks performed by the cell and how modules are placed within the cell.

## 1.3 Limitations

A large variation of cutting tools are used when producing the aircraft components and these cutting tools differ in how they are prepared before they are loaded into the CNC-machines. Some of the tool variants and its tool parameters are similar to each other and these variants will be combined to decrease the amount of variants in the system. This will make system less complex to model whilst still being a sufficient representation of the actual system.

The flexible automation cell is a work-in progress and all the modules are not yet built and tested. Therefore, the data available for these modules is either scarce or not available and how the robot interacts with each module is not yet established. Because of this, extended amounts of time will not be spent trying to gather high quality data.

## 1.4 Research questions

1. To what extent can DES be used as a tool when designing new production systems?

2. Can the simulation model of the flexible automation cell be used to make design decisions to prove the viability of DES in the early design of production systems?

# 2
# Theory

*The following chapter describes the theory of the project. DES will be introduced along with its application within production systems followed by an explanation of input data management. Finally, in order to answer the research questions, theory on the design of production systems will be provided.*

## 2.1 Discrete Event Simulation

By definition, simulation is the act of virtually reproducing a system or process from reality [7]. This can be done both by hand or with the assistance of computers, and the goal of the simulation is to understand and improve its real-world counterpart [6]. It is also used during design phases of systems to gain knowledge of how the system will behave, and with this knowledge, make early design decisions [6].

When creating a simulation, one also creates a simulation model. This model contains the objects within a process or system and the relationship between these objects [6]. DES is applied when modelling a discrete event system, which is often the case when modelling a production system. A discrete event system, as opposed to a continuous system, changes states when a particular event of interest occurs, while a continuous system does not [5], [6].

For example: There is a buffer and a conveyor within a production system. The system changes state when a product is loaded or unloaded, e.g. the amount of products in the buffer goes from one to two or vice versa, and this happens in discrete steps. Compare this to the position of a product on a moving conveyor where the position is continuous, as the position changes in infinitesimally small steps. This is visualised in Figure 2.1.

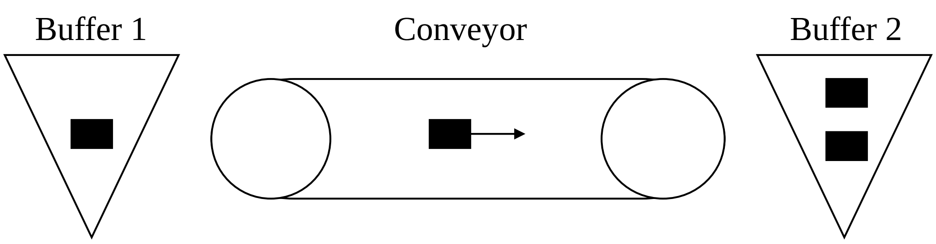



**Figure 2.1:** Figure showing the discrete amount of components in two buffers and the continuous position of a component on a conveyor.

Furthermore, many measurements made in a production system are discrete, such as the amount of products completed in a day, the amount of times a machine breaks down and available workers [5].

Even though DES has many applications, it also has its disadvantages. The following list shows advantages and disadvantages of DES:

**Advantages [5],[6]**

- Virtual tool used to find improvement potential within a system
- Possibility to virtually test modifications of a system without physically interrupting the actual system
- Easier and cheaper to modify a DES model than the actual system when testing improvements
- Can be used to investigate and predict economical benefits of certain modifications that can be made to a system

**Disadvantages [6]**

- Creating a simulation model requires special training and skills
- There are endless ways of modelling a system
- Results from the simulation may be hard to interpret
- Analysing by using a simulation model can be expensive and time-consuming
- DES cannot accurately describe everything in a production systems

The parts within a production system which cannot be accurately captured using only DES, such as machine dynamics, are usually simulated through continuous simulation. If one is interested in accurately representing system dynamics as well, a hybrid simulation where DES interacts with continuous simulation within a hybrid system is a possible approach [8]. Furthermore, there is the concept of *digital twin*, where one creates a digital ”twin” of a real-world system or process [9]. This twin is connected to the same controllers as its real-world counterpart and can behave exactly the same. This also means that the controller can be programmed before it is implemented in the physical system, which is known as *virtual commissioning* [9].

### 2.1.1 Applications of DES in production systems

Historically DES has been used when designing new production systems. However, a shift towards using DES as a tool to simulate daily operations such as maintenance and operation planning is evident [10]. DES is also used in the analysis of the behaviour of systems that are not necessarily defined as a common production system, such as hospitals [11]. However, these type of systems are similar to a regular production system since both of them have a set of tasks that are performed in a certain sequence [11].

## 2.2 Input data management, statistics and distributions

Input data management is defined by Skoogh and Johansson [12] as the process of deciding the required parameters, collecting data, analysing and translating raw data into usable input data and documenting the data. It is also one of the most time consuming and expensive but important steps within DES projects [6], [12]. If the input data is not accurate, it might lead to an inaccurate output which in turn might lead to improvement suggestions that could prove to be mistakes [6]. Therefore it is important to begin gathering data as early as possible in the project and, if possible, gather the data directly from the real system. However, this might prove problematic in cases where the system being modelled is in early design stages [6], [12].

A way to manage the input data is to classify the data by using the classifications suggested by Robinson and Bhatia [13]. Data is divided into three categories: A,B,C, see Table 2.1, to know what quality the data has and how to gather it. Knowing which data type that is used in certain parts of the model is important to know how reliable the results from the model are. The preferred data to be used in a DES is category A data, but it is seldom the case that this data is readily available. In cases where category A data does not exist, one has to collect category B instead. A simple method of collecting category B data is by doing a stop watch time study to collect the necessary data [12]. This however is only possible if the system one wishes to model is fully functional and running. If that is not the case, one has to estimate category C data. As category C data are only estimates mean that the quality of the data is lower than A and B. To aid in the estimation of the required data, one can discuss with experts on the system and production engineers within the company. One can also look for similar systems with similar processes and make estimations with the help of the data from such systems [6], [12].

**Table 2.1:** Classification of data [13]

| | |
|---|---|
| A | Available |
| B | Not available but collectable |
| C | Not available and not collectable |

The raw data gathered has to be translated into useful input data for the DES. In many cases, there is some variability that has to be represented in the DES model and a common way to represent this variability is through statistical distributions [12]. The usable distributions vary depending on the quality of the data and more importantly its availability. According to Banks [6], two distributions that are useful when data is limited or of low quality are *triangular* and *uniform* distributions.

### 2.2.1 Uniform distribution

The uniform distribution is one of the most common and simple statistic distributions. It is used when one has a lower limit $a$ and a upper limit $b$ and a random variable $x$ which can take any value between the $a$ and $b$ limit and the likelihood of getting any of the values is equal [14]. Its probability density function (PDF) is defined in Equation 2.1:

$$f(x) = \begin{cases} \frac{1}{b-a}, & a \leq x \leq b \\ \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

For example: There is limited amount of data about about a machines processing time, but it is somewhere between 2 and 5 minutes. Its lower limit $a$ would be 2, and its upper limit $b$ would be 5. Its PDF is presented in Figure 2.2.

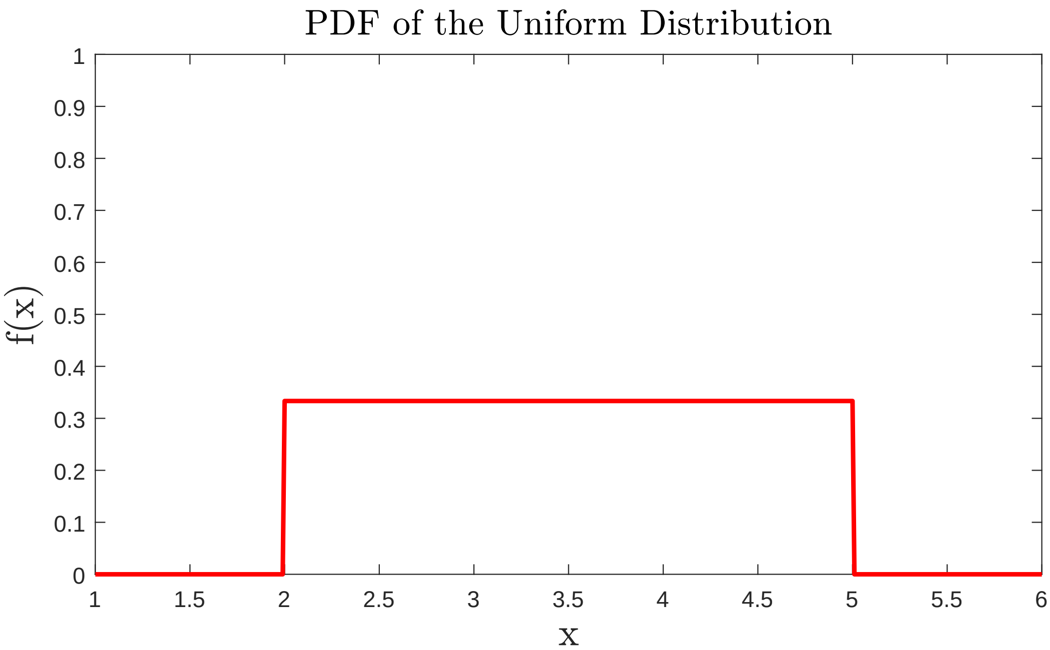



**Figure 2.2:** PDF of the uniform distribution in the machine example.

### 2.2.2 Triangular distribution

The triangular distribution is similar to the uniform distribution, with the exception that one also knows an approximate mean value. The triangular distribution requires more data, but is a substantial improvement to the uniform distribution as there is knowledge of where the values usually fall, making it less random [6]. It has a lower limit $a$, a mean value $b$ and a upper limit $c$. The PDF of the triangular distribution is defined in Equation 2.2.

$$f(x) = \begin{cases} \dfrac{2(x-a)}{(b-a)(c-a)}, & a \leq x \leq b \\ \dfrac{2(c-x)}{(c-b)(c-a)}, & b < x \leq c \\ 0, & \text{elsewhere} \end{cases} \tag{2.2}$$

The machine example from the uniform distribution can be applied as a triangular distribution example. However, this time there is also the mean value $b$, which in this case is given the value 3. Its PDF is presented in Figure 2.3.

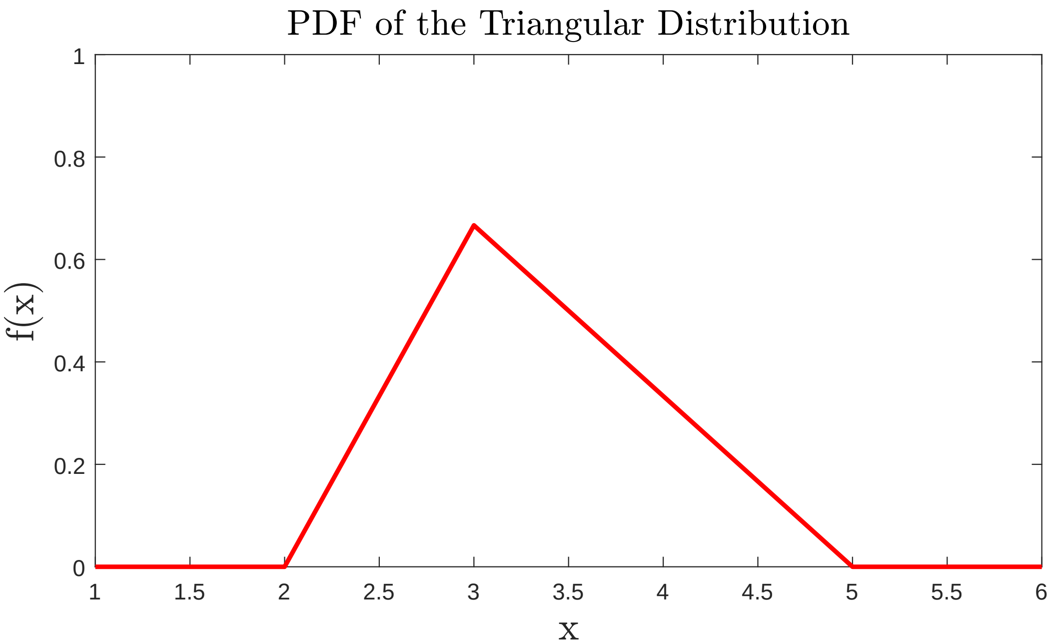



**Figure 2.3:** PDF of the triangular distribution in the machine example.

## 2.3 Design of production systems in early phases

In production system design, it is important that resources are put into the design when the production system is in its early design phases. If the design is not properly done, it can lead to problems when the production system is implemented and later on in its life-cycle [15].

In the manufacturing industry, there is no generally agreed upon way of designing production systems [15]. However, Rösiö and Bruch [15] studied literature related to production system design and created a framework which shows a common way of production system design, see Figure 2.4. In this framework, common activities of early production system design are mapped and spread out on four different phases: Initiation, background study, pre-study and design of conceptual production systems. The framework also shows the iterative approach to production system design, where the design is evaluated in the fourth phase and if the results are not satisfactory, the process is iterated. It is worth noting that system simulation is a common activity when evaluating production system design.

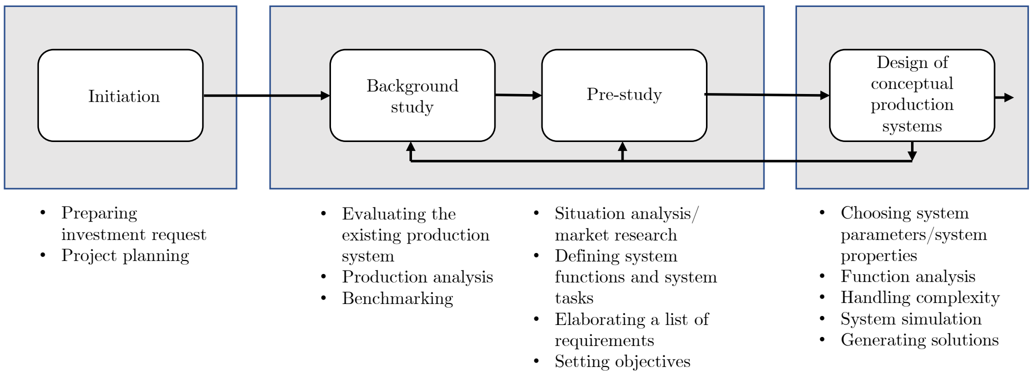



**Figure 2.4:** The four phases of production system design by Rösiö and Bruch [15].

When designing production system layout, Maynard and Zandin [16] presents a layout design tool where one maps the relationship between objects in a production system and weighs the importance of the proximity between these objects. These objects can be machines, storage, work stations, break rooms etc. Using this technique, one can design a production system layout optimally, depending on which objects need to be close to each other and which do not. A common way of evaluating such layouts is by using simulation [16].

In addition to designing the system layout, the flow of the system has to be designed in detail. A common approach is breaking down every process required to produce a product into flow charts. These flow charts are then used to evaluate different solutions and to find the optimal flow of the product from raw material to finished product [17]. The system parameters are connected to the system requirements, and when choosing which parameters the system requires, focus is put on choosing parameters which can compared to already existing data [17].

# 3 Methods

*In this chapter the methods used in the project will be presented. In order to answer the proposed research questions, a a research methodology will be presented which was used to gather relevant qualitative and quantitative data. Thereafter, a DES methodology will be presented which was used during the DES modelling phase. The method will be used to create the DES model and as an extension give the authors knowledge about the design capabilities of DES. Finally, a method in determining the viability of DES will be provided.*

## 3.1 Literature study

To find information related to the subject of using DES as a tool in the early design stages of production systems a literature study was performed. From this literature study, interesting information that other researchers has concluded from their studies such as methodologies used and what challenges that were faced when using DES as a design tool, were found.

The information gathering process was performed by defining important keywords that were used to narrow the scope of the information gathered. The keywords were defined by using past experiences in the subject of DES and in collaboration with the supervisors of the project that has extensive experience in the area. The following keywords were used: *DES*, *Discrete Event Simulation*, *Discrete Event Systems*, *Simulation*, *Virtual Commissioning*, *Digital Twin*, *Statistical Distributions* and *Data Management*. In conjunction with these main keywords, sub-keywords were used to narrow down the results even further, which were: *Viability*, *Challenges*, *Case-study*, *Early Design* and *Production Systems*.

The established keywords were used in the Chalmers library database and the databases Scopus and Google Scholar. To screen the gathered literature, the abstracts of the literature were analysed to decide if they were relevant for further review. The process of reviewing the literature followed the five guidelines suggested by Jones [18]:

- What are the main concepts and ideas of the article?
- Do the ideas and theory align with other literature or do they differ?
- How do you explain the difference?
- Is there any agreements between different literature regarding specific concepts

or do they all differ?
- Which of the concepts and ideas is relevant to your own studies?

## 3.2 Case study on the viability of DES in the early design stages of production systems

To answer the posed research questions, a case study was performed. The case which was studied was the application of DES on the flexible automation cell introduced in Section 1.1. The flexible automation cell was modelled and simulated using Banks simulation methodology, explained in Section 3.3. The results of the simulation was subsequently analysed and used as support for the posed research questions (see Section 1.4).

## 3.3 Banks simulation methodology

A common methodology utilised to create DES models is the methodology suggested by Banks [6]. The structure for the method is presented in Figure 3.1. This structure has been divided into three main phases: **Research & preparation**, **Model building** and **Analysis**.

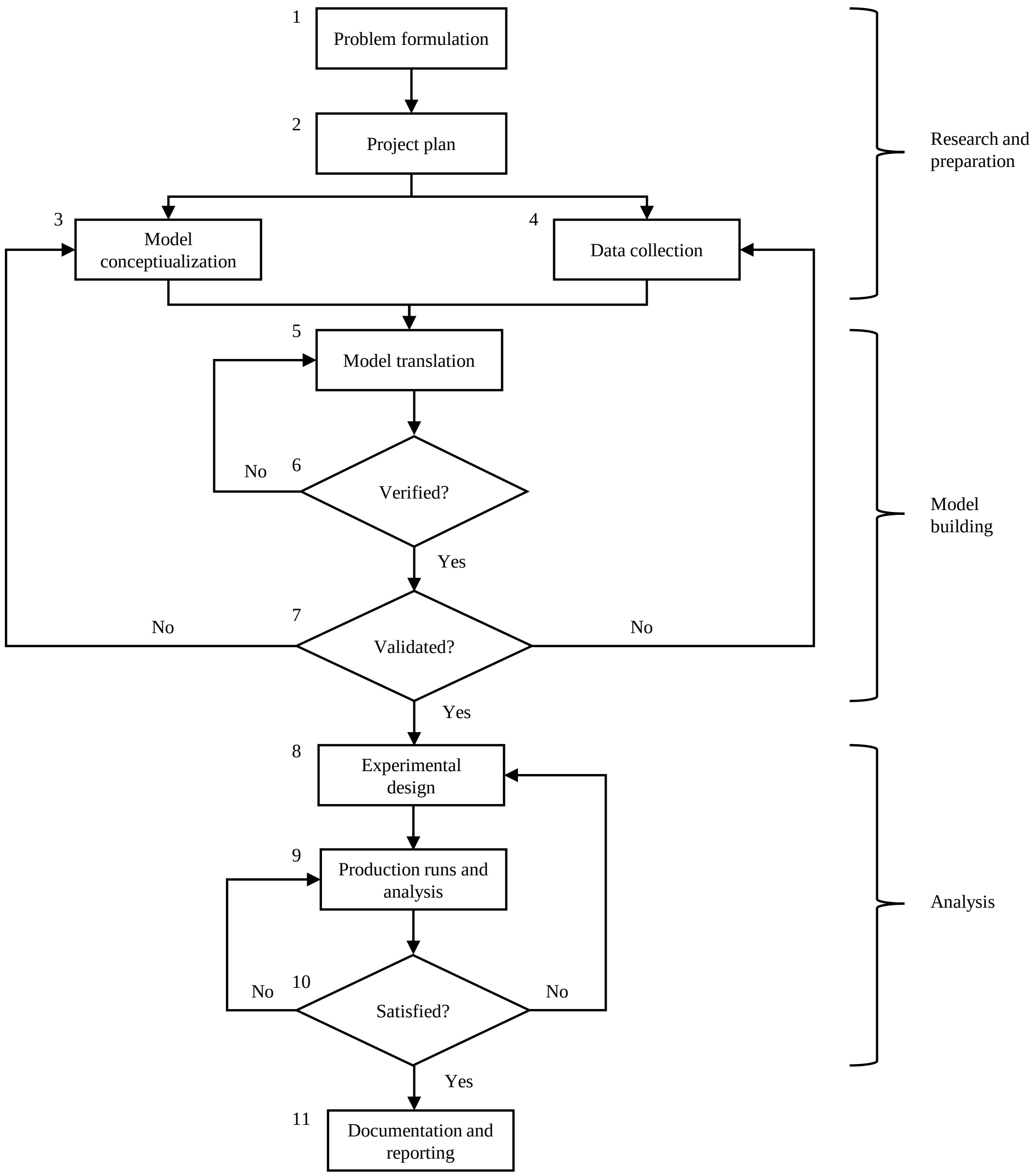



**Figure 3.1:** Work procedure within simulation suggested by Banks [6]

### 3.3.1 Research & preparation

#### 1. Problem formulation

Developing a problem formulation in the early stages of a simulation project is of great importance to ensure that the model and results that are being gathered are aligned with the expectations of the stakeholders [6]. The problem formulation might have already been developed by the stakeholders whereby the modeller has to verify that the problem formulation is adequate for the project [6]. It can also be developed together with the modeller through discussions with the stakeholders.

In the case of this project, the stakeholders had developed a rough problem formulation which was further developed together with the authors, from here on referred as the modellers. This was done through unstructured discussions with the stakeholders which continued until the stakeholders and the modellers had developed a final problem formulation.

#### 2. Project plan

The project is planned by analysing important tasks that has to be performed with regards to the objectives and problem formulation [6]. A project schedule is then created by deciding when certain tasks should be completed.

A project plan was created early in the project to get an overview of the steps required to create a DES model of the system. This project plan was created by analysing which parts of the project were important and further mapped when they should be completed in a Gantt-chart.

#### 3. Model conceptualisation

A model conceptualisation is a way of abstracting the real system and deciding what to model and at what detail level [19]. This abstraction of reality is therefore a simplification of how the system actually behaves. Depending on the purpose of the project, some parts of the system can be modelled with higher detail than others. Different purposes would result in a different abstraction and simplification of the real system, and the results of the simulation would thereby change depending on the purpose of the project [19].

The conceptual model is independent of which software or programming language that is used [19]. It is merely a representation of the logical behaviour within the system such as: which tasks are performed, in what sequence are they performed and data regarding the tasks.

For this project, the model conceptualisation was done by discussing the system behaviour with the stakeholders, especially the developers of the system. The majority of the conceptual model was based on a process map showing every task of the system and their order which was provided by the stakeholders. This process map was translated into a sequential function chart (SFC) showing states, actions/-operations and guards/conditions of the system (see Fig 3.2), explained by Battikha [20], to simplify the translation into an operational model.

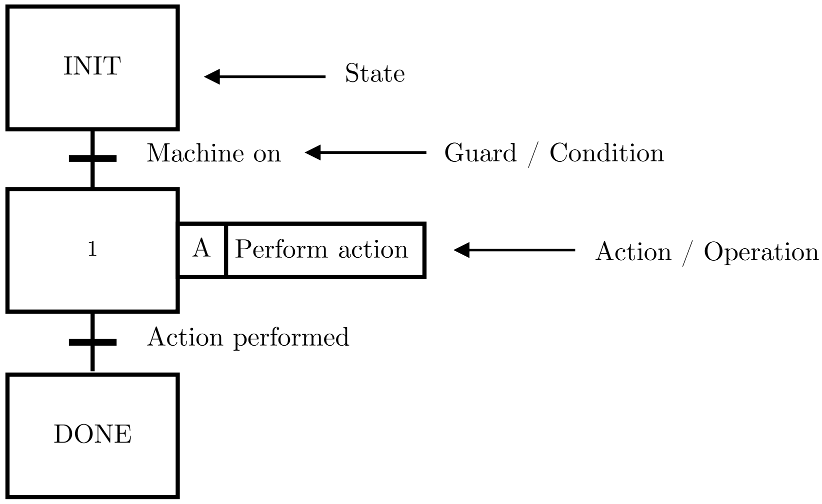



**Figure 3.2:** Explanation of a SFC

**4. Data collection**

Input data management is important to achieve a precise model that can replicate the behaviour of a system. Managing the input data and gathering the data is one of the most time-consuming processes in a DES project [12]. Before the actual data collection can begin, one has to define what data is required in the model. Examples of data parameters that are commonly required for a DES model are process times, cycle times, setup times and availability. The decision on which parameters to be used is decided by the modeller together with personnel who are experienced with the system [12]. Data management was explained more in depth in Section 2.2

The data collection phase consisted of firstly finding which parameters that the operational model would require. These parameters were decided by the modellers during the model conceptualisation phase and compiled into a list, see Table 4.2. This list was passed on to the supervisor of the project, whereby the supervisor checked with the relevant parties for the availability of data for the listed parameters. If there was no data available for a certain parameter, a stopwatch time study was conducted if the data was gatherable. A stopwatch study is conducted by measuring the time it takes for an operation performed by an operator (or robot) to be completed [16].

The data required from the CNC-machines was gathered by the relevant production engineers and supplied to the modellers. This was also the case for the worker data required.

The time it takes for the robot to travel from one module position to another was gathered by simulating the actual robot program in the software RobotStudio by

ABB Robotics. Every possible movement was timed and compiled into a data table which was subsequently transferred to the simulation software.

As the flexible automation cell will perform operations that currently are performed manually, the operators performing the manual operations were observed and timed to achieve a more accurate data estimation. Firstly, a sample size was determined to be confident that the data gathered was accurate. Thereafter, the processes were timed with three different operators performing the processes. Two of the operators were beginners (the modellers) and one was experienced. The processes were broken down into loading time, processing time and unloading time.

If the data was not gatherable, as with the case of the module data, the modellers held unstructured interviews with experts of the system to make estimations of the required data as suggested by Banks [6]. This was combined with the manual tool preparation data to make the estimated module data more accurate.

The data gathered was fit to appropriate statistical distributions, i.e triangular distribution, with the use of MATLAB for the data to be usable in the simulation software.

### 3.3.2 Model building

The model building phase is performed by translating the conceptual model into an operational model (computerised model). One has to verify that the operational model is accurate to the conceptual model and finally validate that the operational model is accurate to the real system [6].

#### 5. Model translation

The model translation step is where the conceptual model is translated into an operational model. The conceptual model will be translated using a chosen simulation software, such as AutoMod and Tecnomatix Plant Simulation, or programming language. The operational model is continuously verified and validated by comparing it to the conceptual model and, if possible, the real system [6].

The DES modelling software chosen for the project was Tecnomatix Plant Simulation, which was already in use at GKN Aerospace and as such was the preferred alternative. As there was no prior knowledge about Tecnomatix Plant Simulation within the project, the modellers gained the required knowledge through educational material provided by Siemens, whom owns Tecnomatix, and by asking specific questions at the Plant Simulation forums. Furthermore, a Plant Simulation expert from Siemens was consulted throughout the project. The conceptual model created in step 3 was then translated into a operational model by following the SFCs created and by continuously asking the system experts whether the system was behaving as intended when a new part of the system logic was translated into the operational model.

## 6. Verification

The operational model has to be accurate to the conceptual model, and as such it is important to verify it. The operational model is verified by comparing how accurate its behaviour is with respect to the conceptual model. To ensure that it behaves correctly, the code in the operational model is often debugged extensively whereby each new addition of the model logic is thoroughly checked if it is behaving correctly. If the operational model behaves as intended when compared to the conceptual model, it is considered to be verified [6]. Another important part of the verification of the operational model is to thoroughly document the code. By having everything documented with explanations of variables and objects, the model can be more easily verified by someone that is not the original model builder or if the model builder has to verify older code [6].

The operational model in this project was verified by extensively debugging every part of the translated logic. This was done using the debugging tool built into Tecnomatix Plant Simulation and continuously monitoring certain parameters of interest when simulating, such as product destinations and robot state, to make sure that they had reasonable values. If any part of the logic did not behave according to the conceptual model, the problem was located through debugging and fixed through iteration. System experts were consulted to verify if the operational model was behaving as the system was supposed to.

## 7. Validation

The validation process is where the verified operational model is compared to the real system. The method of validating the operational model is done through iteration, where the model is simulated, the behaviour compared to reality, the model and its parameters adjusted and the process is iterated. This process is iterated until the operational model is sufficiently accurate to the real system [6].

First of all, the model should look realistic to the user. By involving the user of the model, one gains important input as to which aspects of the model look accurate to reality and which do not. If the model when compared to the real system looks correct for the user, the model builder can be confident in that the model is accurate to reality, this is called face validation [6]. Another common method of validation is by comparing the output of the model to historical data and from this one can gain an appreciation of the accuracy of the model [6]. Furthermore, the use of a Turing test to validate the model can be a sufficient method of validation. A Turing test is when the output of the model cannot be distinguished from the output of the real system [6]. Banks [6] suggests that one generates an amount of fake output reports and mixes them with real reports. A set of engineers will then try to figure out which reports are fake and which are real, and if the engineers are not able to tell which are the fake reports it means that the model has passed the Turing test.

The method used for validating the model in this project was face validation. The finished operational model was showed to the user and system experts, and if something didn't work as they had intended it was changed according to their feedback. This process was repeated until they were satisfied with the behaviour and look of the model and as such was deemed validated.

### 3.3.3 Analysis

The analysis part of the simulation is where one simulates and analyses the results to find potential bottlenecks within the system. In an effort to break these bottlenecks, the **Theory of Constraints** (TOC) developed by Eliyahu M. Goldratt [1] is a well-known method. Using this method, one can find the bottleneck(s) using the simulation, try different ideas on how to break the bottleneck(s) and then iterate until the bottleneck is broken.

**8. Experimental design**

During the experimental design part of the project, the modeller determines the run length, number of runs to be made and length of a potential warm-up period for every variation of the model. When the modeller is satisfied with the results of a certain model variant, the model can be tweaked and the process starts over again [6].

The experimental designs of the model were decided together with the stakeholders where the stakeholders explained which experimental models would be of interest. The operational model was changed according to the stakeholders wishes and simulated until the modellers and stakeholder were satisfied with the result. Initially each experiment is subject to a screening process where they are simulated with shorter run lengths and fever number of runs to determine if they impact the system at all. This initial screening process of experiments was performed to be more time efficient.

**9. Production runs and analysis**

The result of the simulation can be analysed to find potential improvements to the model. Many simulation software solutions have built-in statistical analysis tools which can be used during the analysis stage of the DES. This analysis can be used to find bottleneck(s) and other potential improvement areas within the model, and subsequently the real life system if the improvements are considered for real life implementation [6].

The operational model and the experimental models were analysed using the built in statistical analysis tools within Tecnomatix Plant Simulation. These tools were *BottleneckAnalyzer*, which visualises the statistical flow of objects within the system through data tables, and *ExperimentManager*, which is used to experiment with different parameters within the system to find the optimal parameter configurations. The *BottleneckAnalyzer* tool was used to find the bottleneck of the system by analysing the percentage of time that the individual modules were working, blocked and starved. A resource which is working the majority of the time, which has a starved resource upstream and a blocked resource downstream, is likely a bottleneck of the system. This bottleneck detection method is called the **Arrow Method** (AM) [21].

Attempts were made to break the identified bottleneck using the *ExperimentManager* tool in conjuncture with the TOC. During the bottleneck breaking phase of TOC, where one tries different ideas of how to break the bottleneck, the *ExperimentManager* was utilised to run experimental runs with experimental parameters in an effort to break the bottleneck(s). This process was repeated until the modellers were satisfied with the results.

#### 10. Satisfied?

If one is satisfied with the results and deem that more production runs are unnecessary, one can proceed to the next step. However, if the results are not good enough, the modeller can do more runs, either by experimenting further with the design or only running the simulation more times to gather more data [6].

#### 11. Documentation and reporting

Lastly, documentation of the project is of great importance. Both by documenting the actual process of the simulation project and documenting the code of the model. Having the process documented, one can gain inspiration in future projects by looking at the documentation of the process [6]. Furthermore, by documenting the code of the model it can be more easily understood by people that are not involved with the project and it can also be of help when the modeller is looking and trying to understand older code [6].

The DES project was documented using the online text-editor Overleaf where the process was thoroughly explained and reported. In addition, the code written in Tecnomatix Plant Simulation was commented for the user to easily understand the system logic and for the modellers to easily go back and understand older versions of the code when necessary.

## 3.4 Determining the viability of DES

To determine the viability of DES in early design stages, a scoring system was created which scores the different steps of the model building process. These scores are based on the difficulty of completing the individual steps with adequate results within the time frame of the project. The sheet is to be filled out by every member of the DES modelling group individually and subsequently discussed to reach a consensus.

For the results generated to not be skewed, one must also consider the fact that errors and improvement potentials can be discovered during any of the modelling steps and not only from the simulation results. As such, any errors or improvement suggestions discovered will be taken into consideration into the final scoring. Taking this into consideration, the modellers created a DES viability worksheet based on the steps in Banks methodology. The questions for each step of the Banks methodology in the viability sheet was influenced by the challenges of DES modelling which was established by the literature study, the challenges can be seen in Table 4.1. Additional challenges related to DES modelling was suggested by Banks [6]. For certain steps of the Banks methodology where challenges are common, additional questions are included in the sheet, e.g. data collection and validation.

The individual steps in the viability sheet is scored from 1-5 where 5 is the highest score and 1 is the lowest. The scoring criteria used to evaluate each step is shown in Table 3.1. Finally, the scoring of all the individual steps are summarised to get a final score. To evaluate the impact of the final score, four different project outcomes was established and each outcome has a certain range of scoring related to it. E.g to concluded that DES was viable for the project, a final score ranging between 76-100 had to be achieved, shown in Figure 3.3. The range of scoring used to evaluate the project outcomes was constructed by having equal ranges for the four project outcomes.

This worksheet was used to gain perspective on the viability of DES when considering the actual project. The final worksheet is presented in Figure 3.3 and contains different questions depending on the individual steps within Banks methodology, a score concerning each question and if there any additional comments. The creation of this worksheet was inspired by different ergonomic evaluation worksheets, such as Rapid Entire Body Assessment (REBA) and Rapid Upper Limb Assessment (RULA) where one scores the ergonomics in a certain position from bad to good ergonomic strain [22]. With the completion of the worksheet one receives a final score and whether action should be taken to improve the ergonomics or not [22].

**Figure 3.3:** DES viability worksheet based on Banks method

**Table 3.1:** The scoring criteria used for the viability sheet

| Score | Description |
|---|---|
| 5 | Very Accurate |
| 4 | Accurate |
| 3 | Moderately Accurate |
| 2 | Inaccurate |
| 1 | Very Inaccurate |

# 4

# Results

*Firstly, the results of literature study which was performed to find articles relevant to the modellers work and the posed research questions, are provided. Secondly, the results from the case study are provided to give the reader a view into the viability of DES in the early design of production systems. Thereafter, the DES viability sheet results are shown. Finally, the results are summarised and the research questions are answered.*

## 4.1 Literature study results

Florecs-Garca *et al.*[23] performed three case studies and mapped recurring challenges in the design, development and deployment phases of DES modelling projects, which is presented in Table 4.1. While all challenges present in Table 4.1 are critical, there is one challenge that proved to be more critical than the rest in each phase [23]. In the design phase, the most critical challenge is defining the problems in the production system and translating the same problems into a DES model. This challenge originates from the differing views on the production system as a whole by different team members. All the cases had to spend a considerable amount of time in the early design stages deciding on what parts of the production system were required for the DES model and on what abstraction level it should be [23].

In the development phase, the most critical challenge was the input data collection, more specifically the lack of existing required data. This challenge becomes the most critical as there is no real life production system to actually gather data from, leading the teams in the cases to estimate data for the DES model [23], which according to Skoogh *et al.* [10] is category C data, and thus of low quality. An interesting difference between the cases is that two of the teams decided on the required input parameters and collected data from that requirement, while the other one collected all available data only to find that some of the data was not required for the DES model [23]. This shows the importance of compiling a list of the data parameters required before data gathering begins.

One of the grandest challenges of the deployment phase was the fact that a majority of the people at the different manufacturing companies did not have any knowledge of DES. This led to a lot of time being spent waiting for the support of the simulation specialists within the cases. The lack of DES knowledge also led to the results being hard to interpret for the rest of the team members, which made decision making based on the DES results challenging [23].

**Table 4.1:** Table showing the common challenges in a DES modelling project by Florecs-Garca *et al.*[23].

| DES modelling challenges | |
|---|---|
| DES modelling phase | Challenge |
| Design | • Decision support restricted by question-specific model formulation. What is the problem and how is it addressed?<br>• Representation of production system dynamics and complexity<br>• Validity of model detail level<br>• Simplification of production system complexity and factor interdependence<br>• Non-uniform abstraction level for model simplification<br>• Modelling combinatorial explosion of options in a production process<br>• Incomplete and conflicting production system knowledge<br>• Software diversity and lack of standardisation |
| Development | • Model verification and validation<br>• Model development time<br>• Input data collection and analysis<br>• Input data availability and quality |
| Deployment | • Model interopability and information sharing between models<br>• Industry acceptance of DES<br>• Communication of results for effective decision making<br>• Simulation model maintenance<br>• Consideration of trade-offs and non-intuitive decisions<br>• High cost and low reusability of models |

A case study done by Freiberg and Scholz [24] examined whether DES could be used to evaluate the effectiveness of modern manufacturing equipment in an existing production system. A DES model based on the existing factory was created using Siemens Plant Simulation and compared to an experimental model containing modern manufacturing equipment replacing the old equipment [24]. The experimental model performed better than the existing factory in terms of product throughput, and as such was deemed a worthy investment purely based on profit [24]. Freiberg and Scholz [24] also mentioned the other benefits of a DES model, such as it providing a good overview of the performance of the different machines in the existing system, the overall performance of the system and that of the future system [24].

The success of the case by Freiberg and Scholz [24] provides an example of the viability of DES in early design stages of production systems even though the experimental model in the case is based on an existing production system.

## 4.2 Case study results

To answer the research questions posed in Section 1.4, the results from the completed case study will be presented and explained. The results will provide insight into the case, which will used to determine the viability of DES in the early design of production systems.

### 4.2.1 Conceptual model

The modelled system was divided into two product lines: the CNC-machines and the flexible automation cell, illustrated in Figure 4.1. The CNC-machines feeds the flexible automation cell with used tools that the cell should prepare and replace. These tools are transported from the CNC-machines to the cell by human operators.

The cell consists of 10 positions where 10 individual modules or buffers can be located. Each module has a specific task which it carries out in the process of preparing the tools. However, not all tool types visit every module, as some modules process specific tool types. The tools are moved from module to module by the robot, which is equipped with two grippers and can carry two tools at the same time.

Currently, 6 of 8 modules can perform their task without the assistance of the robot, thus the robot can perform other tasks whilst such a module is processing a tool. For the other two modules the robot is occupied during its process and cannot perform additional tasks. Furthermore, one of the modules requires a human operator to run it. This module is to be used as a backup for the automatic modules. Should a module break, the tool will be placed in the manual interaction module and the tool is prepared manually by the operator and placed back in the system.

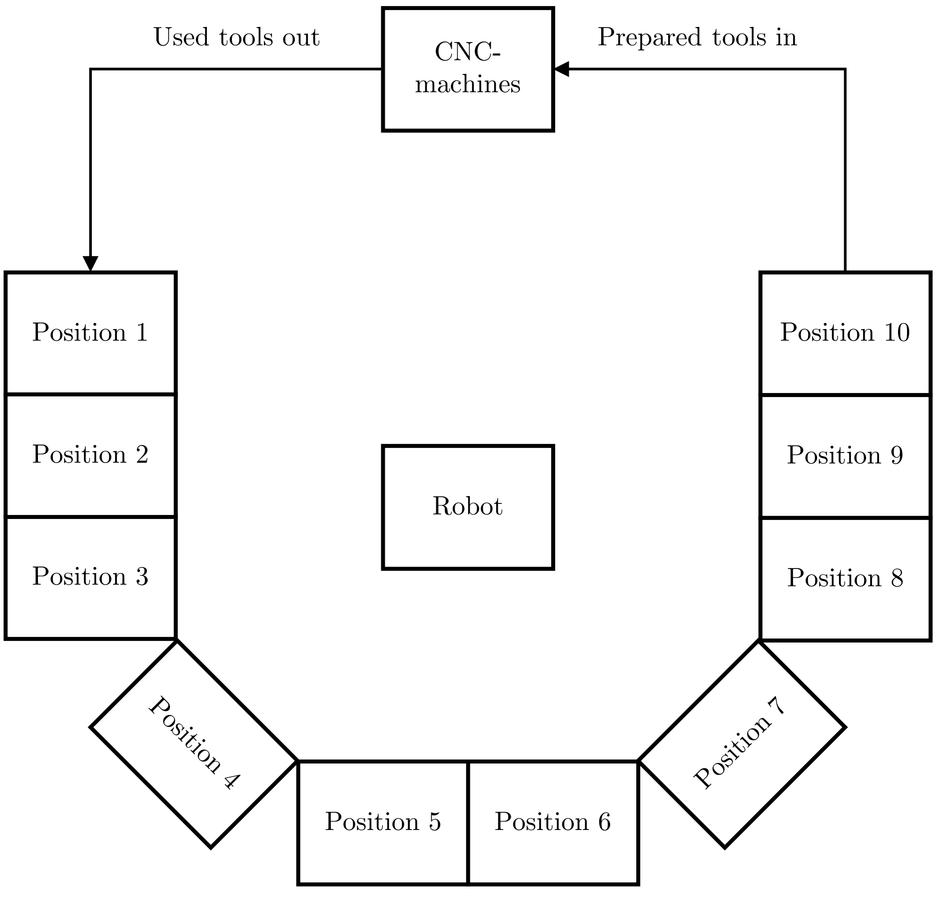



**Figure 4.1:** Conceptual model showing the main flow of the system

### 4.2.2 Data collection

#### 4.2.2.1 Data parameters

The parameters that the modellers decided were required for the operational model is presented in Table 4.2.

**Table 4.2:** Table showing the data parameters required for the DES model.

| Data parameters required | |
|---|---|
| Operation | Unit |
| CNC-machine loading time | [min] |
| CNC-machine unloading time | [min] |
| CNC-machine processing time | [min] |
| Amount of available workers | |
| Total amount of time the operators work during a day | [h] |
| Time the robot requires to move from one module to another | [s] |
| Time the robot requires to move into a module | [s] |
| Time the robot requires to move out of a module | [s] |
| Manual solid tool preparation loading time | [s] |
| Manual solid tool preparation unloading time | [s] |
| Manual solid tool preparation processing time | [s] |
| Manual insert preparation loading time | [s] |
| Manual insert preparation unloading time | [s] |
| Manual insert preparation processing time | [s] |
| Manual collet preparation loading time | [s] |
| Manual collet preparation unloading time | [s] |
| Manual collet preparation processing time | [s] |
| In/Out module processing time | [s] |
| Cleaning module processing time | [s] |
| Vision module processing time | [s] |
| Solid tool change module processing time | [s] |
| Insert change module processing time | [s] |
| Collet change module processing time | [s] |

#### 4.2.2.2 CNC-machine and worker data

The unloading, loading and processing time of the CNC-machines all depend on the specific product in the machine as well as the current tempo. The data gathered is presented in Table 4.3 and Table 4.4. There are five operators available, with two shifts and a total of 18 hours per day.

**Table 4.3:** Table showing the unloading, loading and processing times of tempo 1 of the different products processed in the CNC-machines.

| CNC-machine data for tempo 1 [min] | | | |
|---|---|---|---|
| Product | Unloading time tempo 1 | Loading time tempo 1 | Processing time tempo 1 |
| A | 5.00 | 5.00 | 254.12 |
| B | 4.25 | 4.25 | 347.73 |
| C | 4.50 | 4.50 | 226.57 |
| D | 2.00 | 2.00 | 45.00 |
| E | 5.25 | 5.25 | 252.00 |
| F | 8.00 | 8.00 | 470.20 |
| G | 8.00 | 8.00 | 470.20 |
| H | 7.25 | 7.25 | 453.91 |

**Table 4.4:** Table showing the unloading, loading and processing times of tempo 2 of the different products processed in the CNC-machines.

| CNC-machine data for tempo 2 [min] | | | |
|---|---|---|---|
| Product | Unloading time tempo 2 | Loading time tempo 2 | Processing time tempo 2 |
| A | 3.50 | 3.50 | 177.88 |
| B | 1.75 | 1.75 | 143.18 |
| C | 5.25 | 5.25 | 264.37 |
| D | - | - | - |
| E | 3.75 | 3.75 | 180.00 |
| F | 4.25 | 4.25 | 249.80 |
| G | 4.25 | 4.25 | 249.80 |
| H | 4.25 | 4.25 | 266.09 |

#### 4.2.2.3 Robot movement

In Table 4.5 the data gathered from the simulation of the flexible automation cell robot in RobotStudio is presented. The table presents the time it takes in seconds for the robot to move from a specific module position (see Figure 4.1 for the module positions) to any other module position as well as from the robots home position to any module position. The speed of the robot in the simulation is the same as the real-life robot, which has been set by a robotics expert. The time it takes for the robot to move into and out of a module was timed at the real-life cell to be a constant 15$s$ into a module and 15$s$ out of a module.

**Table 4.5:** Table showing the time it takes for the robot to move from position to position in seconds.

| | | To [s] | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Home | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 |
| From | Home | 0.0 | 1.5 | 1.2 | 1 | 0.5 | 0.4 | 0.4 | 0.5 | 0.8 | 1.1 | 1.4 |
| | P1 | 1.5 | 0.0 | 0.5 | 0.8 | 1 | 1.3 | 1.6 | 1.9 | 2.2 | 2.5 | 2.8 |
| | P2 | 1.2 | 0.5 | 0.0 | 0.5 | 0.8 | 1 | 1.3 | 1.6 | 1.9 | 2.2 | 2.5 |
| | P3 | 1 | 0.8 | 0.5 | 0.0 | 0.4 | 0.6 | 1 | 1.2 | 1.5 | 1.8 | 2.2 |
| | P4 | 0.5 | 1 | 0.8 | 0.5 | 0.0 | 0.4 | 0.8 | 1 | 1.2 | 1.6 | 1.9 |
| | P5 | 0.4 | 1.3 | 1 | 0.7 | 0.4 | 0.0 | 0.6 | 0.7 | 1 | 1.3 | 1.6 |
| | P6 | 0.4 | 1.6 | 1.3 | 1 | 0.8 | 0.6 | 0.0 | 0.4 | 0.6 | 0.9 | 1.3 |
| | P7 | 0.5 | 1.9 | 1.6 | 1.2 | 1 | 0.7 | 0.4 | 0.0 | 0.4 | 0.7 | 1 |
| | P8 | 0.8 | 2.2 | 1.9 | 1.5 | 1.2 | 1 | 0.6 | 0.4 | 0.0 | 0.5 | 0.8 |
| | P9 | 1.1 | 2.5 | 2.2 | 1.8 | 1.6 | 1.3 | 0.9 | 0.7 | 0.5 | 0.0 | 0.5 |
| | P10 | 1.4 | 2.8 | 2.5 | 2.2 | 1.9 | 1.6 | 1.3 | 1 | 0.8 | 0.5 | 0.0 |

#### 4.2.2.4 Manual tool preparation

Two different kind of tools are used in the CNC-machines: Tools with solid cutting tools and tools with inserts. In the solid cutting tools, there are also collets which have to be replaced. These collets are not changed every time a solid tool is prepared, but approximately once in 300 preparations. In Table 4.6 the data gathered by observing the three operators performing the manual solid tool change is presented, where operator 1 is experienced and operator 2 and 3 are beginners. The sample size was a total of 15, where 5 experiments were performed by each operator. From this data, an appropriate distribution was chosen, which was the triangular distribution explained in Section 2.2.2. From the collected data, the lowest processing time of $20s$ and the highest processing time of $32s$ were observed and the mean value b was calculated to be $25s$. This resulted in the PDF plotted in MATLAB that is presented in Figure 4.2. The loading time and unloading time were both constantly around $2s$ and was therefore rounded to a value of $2s$.

**Table 4.6:** Table showing the data collected by observing the three operators performing the manual solid tool change.

| Process: Manual Solid Tool Change | | | | |
|---|---|---|---|---|
| Experiment no. | Operator | Loading time [s] | Processing time [s] | Unloading time [s] |
| 1 | 1 | 2 | 20 | 3 |
| 2 | 1 | 2 | 25 | 2 |
| 3 | 1 | 1 | 27 | 1 |
| 4 | 1 | 2 | 24 | 2 |
| 5 | 1 | 2 | 23 | 2 |
| 6 | 2 | 2 | 24 | 2 |
| 7 | 2 | 2 | 30 | 2 |
| 8 | 2 | 2 | 22 | 2 |
| 9 | 2 | 3 | 23 | 2 |
| 10 | 2 | 2 | 32 | 2 |
| 11 | 3 | 2 | 26 | 1 |
| 12 | 3 | 2 | 24 | 2 |
| 13 | 3 | 2 | 25 | 2 |
| 14 | 3 | 2 | 24 | 2 |
| 15 | 3 | 2 | 25 | 2 |

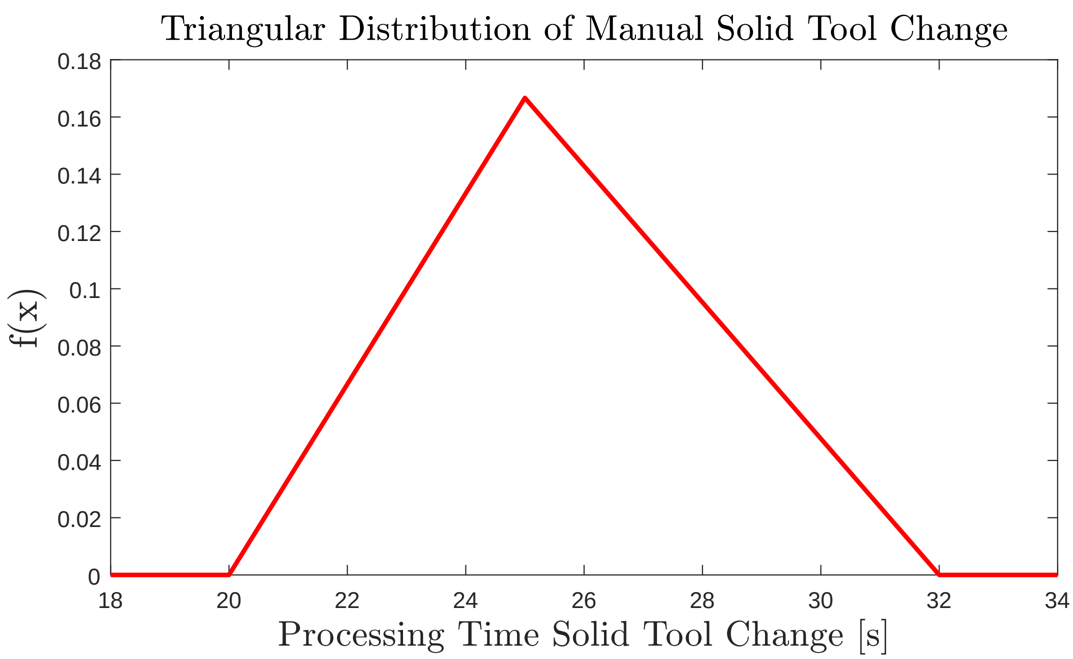



**Figure 4.2:** Triangular distribution of the manual solid change.

In Table 4.7 the data gathered by observing the three operators performing a manual change of one insert is presented. The sample size was 30 in total, with 10 experiments performed per operator. Once again the triangular distribution was used. From the collected data, the lowest processing time of $10s$ and the highest processing time of $38s$ were observed and the mean value b was calculated to be $20s$. This resulted in the PDF that is presented in Figure 4.3. The loading time and unloading time both had a rounded value of a constant $2s$.

**Table 4.7:** Table showing the data collected by observing the three operators performing the manual insert change.

| Process: Manual Insert Change | | | | |
|---|---|---|---|---|
| Experiment no. | Operator | Loading time [s] | Processing time [s] | Unloading time [s] |
| 1 | 1 | 2 | 12 | 2 |
| 2 | 1 | 2 | 14 | 2 |
| 3 | 1 | 2 | 10 | 2 |
| 4 | 1 | 2 | 13 | 2 |
| 5 | 1 | 2 | 10 | 2 |
| 6 | 1 | 2 | 18 | 2 |
| 7 | 1 | 2 | 18 | 2 |
| 8 | 1 | 2 | 16 | 2 |
| 9 | 1 | 2 | 20 | 2 |
| 10 | 1 | 2 | 15 | 2 |
| 11 | 2 | 2 | 27 | 2 |
| 12 | 2 | 2 | 21 | 2 |
| 13 | 2 | 2 | 21 | 2 |
| 14 | 2 | 2 | 25 | 2 |
| 15 | 2 | 2 | 25 | 2 |
| 16 | 2 | 2 | 13 | 2 |
| 17 | 2 | 2 | 14 | 2 |
| 18 | 2 | 2 | 19 | 2 |
| 19 | 2 | 2 | 17 | 2 |
| 20 | 2 | 2 | 13 | 2 |
| 21 | 3 | 2 | 15 | 2 |
| 22 | 3 | 2 | 15 | 2 |
| 23 | 3 | 2 | 25 | 2 |
| 24 | 3 | 2 | 19 | 2 |
| 25 | 3 | 2 | 17 | 2 |
| 26 | 3 | 2 | 32 | 2 |
| 27 | 3 | 2 | 38 | 2 |
| 28 | 3 | 2 | 33 | 2 |
| 29 | 3 | 2 | 28 | 2 |
| 30 | 3 | 2 | 33 | 2 |

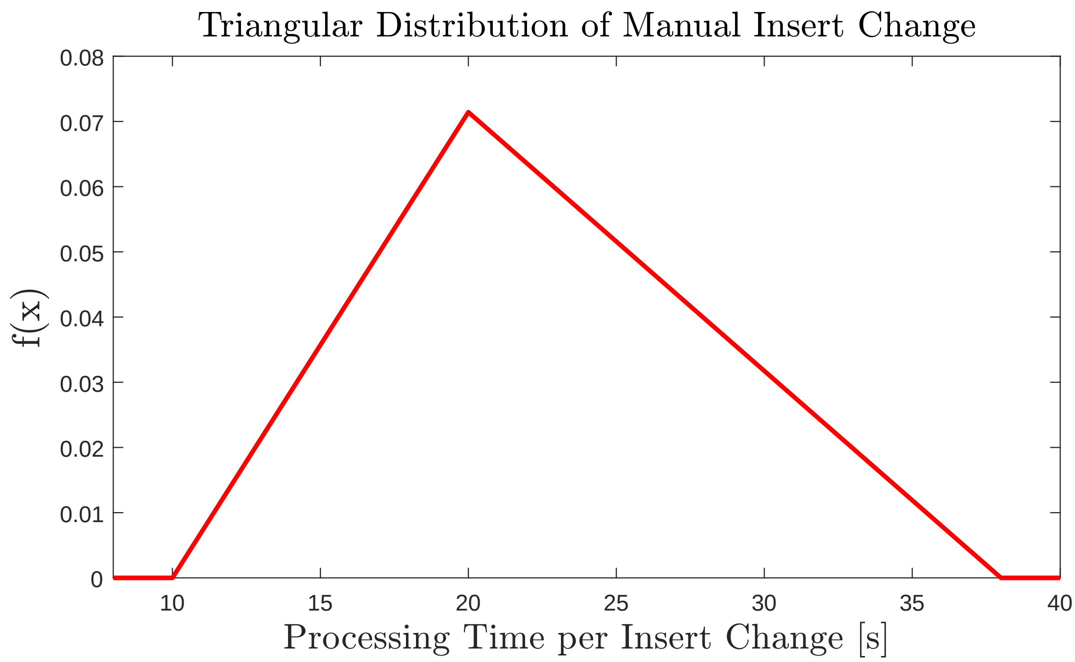



**Figure 4.3:** Triangular distribution of the manual insert change.

In Table 4.8 the data gathered by observing the three operators performing the manual collet change is presented. The sample size was 15 in total, with 5 experiments performed per operator. Once again the triangular distribution was used. From the collected data, the lowest processing time of $22s$ and the highest processing time of $40s$ were observed and the mean value b was calculated to be $30s$. This resulted in the PDF that is presented in Figure 4.4. The loading time was rounded to a constant $4s$ and the unloading time was rounded to a constant $2s$.

**Table 4.8:** Table showing the data collected by observing the three operators performing the manual insert change.

| Process: Manual Collet Change | | | | |
|---|---|---|---|---|
| Experiment no. | Operator | Loading time [s] | Processing time [s] | Unloading time [s] |
| 1 | 1 | 3 | 24 | 2 |
| 2 | 1 | 3 | 26 | 2 |
| 3 | 1 | 3 | 22 | 2 |
| 4 | 1 | 3 | 24 | 2 |
| 5 | 1 | 4 | 24 | 2 |
| 6 | 2 | 5 | 34 | 2 |
| 7 | 2 | 5 | 33 | 2 |
| 8 | 2 | 3 | 28 | 2 |
| 9 | 2 | 3 | 28 | 2 |
| 10 | 2 | 5 | 34 | 2 |
| 11 | 3 | 4 | 34 | 2 |
| 12 | 3 | 3 | 40 | 2 |
| 13 | 3 | 4 | 36 | 2 |
| 14 | 3 | 3 | 35 | 2 |
| 15 | 3 | 3 | 32 | 2 |

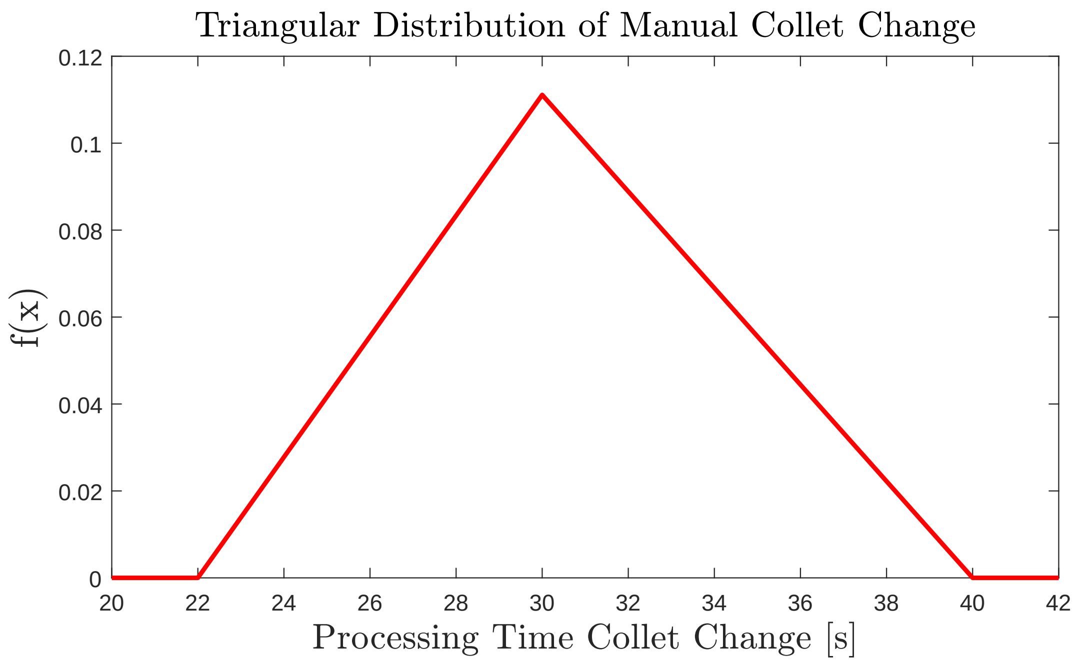



**Figure 4.4:** Triangular distribution of the manual collet change.

#### 4.2.2.5 Module data

When it came to modules which had no real-life operation, system experts were consulted to estimate the required data. The modules which had real-life operations, e.g. solid, insert and collet change, the data from the manual preparation was combined with expert interviews. The estimated module data is presented in Table 4.9.

**Table 4.9:** Table showing the estimated processing times of the modules, with the exception of the manual preparation module.

| Module data | |
|---|---|
| Module | Processing time [s] |
| In/Out: In | 4 |
| Cleaning | 60 |
| Vision | 10 |
| Solid tool change | 25 |
| Insert change | 20 |
| Collet change | 30 |
| In/Out: Out | 4 |
| Manual interaction | See manual preparation |

### 4.2.3 Operational model

The model contains 20 CNC-machines, 10 modules, buffers and a robot. Tools are delivered from the CNC-machines to the input buffer at the flexible automation cell manually by operators. Once the tools are in the flexible automation cell, the robot handles the transportation of tools between the modules. A booking system is utilised where each module sends requests to the robot for it to pick a tool and send it to another module.

When a tool has been processed in the cell it is transported to a warehouse/buffer by an operator. When all the required tools are present in the warehouse, an order is sent by a machine its process starts. A picture of the model is presented in Figure 4.5.

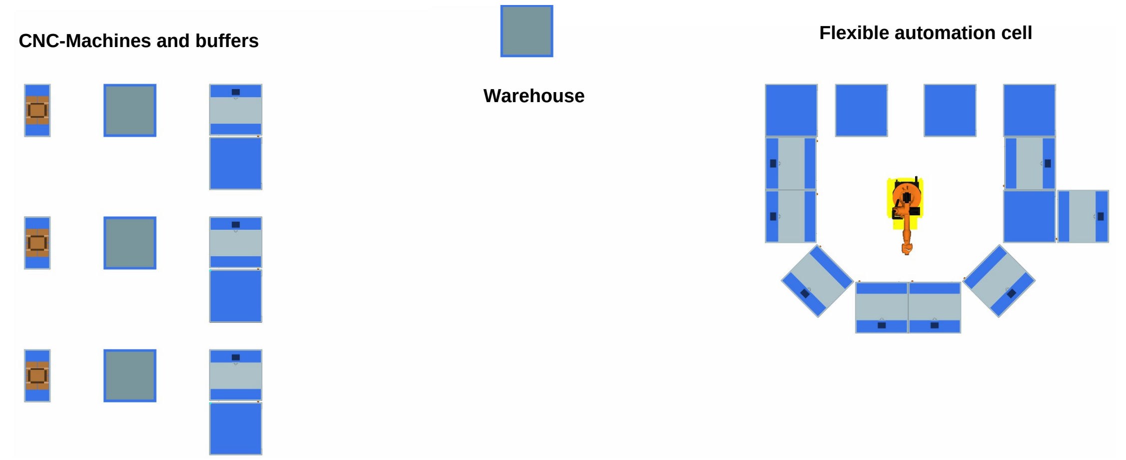



**Figure 4.5:** CNC-machines located to the left, the warehouse at the top and the flexible automation cell to the right.

### 4.2.4 Verification and validation

The operational model was considered verified as a result of debugging and stepping through the simulation, continuously comparing it to the conceptual model. However, the model was not considered completely validated, as there is no real-world system or data of which to compare it to. It was deemed semi-validated, as the system experts deemed that the model behaved as expected.

### 4.2.5 Experimental design

To answer research question 1, see Section 1.4, results from the simulation experiments are explained in detail.

The run length of each experiment was set to 60 days and each experiment was observed five times. When the software is observing an experiment multiple times, the random variables (random seeds) changes and the results of the model may change since there is a larger variance in the model.

Furthermore, the model has to be stable throughout long periods of time. To make sure that the model was stable an initial experiment was performed: the model was run for one year with five observations. Since there was no errors occurring during that time the model was deemed stable.

Additional experiments performed was categorised into two types of experiments: structural and improvement of parameters. The structural experiments changes the behaviour of the system by alternating features such as how parts are delivered to the cell and how the robot operates. These experiments included the use of Automated Guided Vehicles (AGV), having modules be completely self-sustainable and controlling which tool type arrives to the cell and when. The structural experiments performed is presented in Table 4.10. The improvement of parameters experiments focuses on running multi-level experiments to determine what parameter values should be used to improve the output of the cell. Examples of parameters is robot speed and buffer sizes throughout the production.

**Table 4.10:** The structural experiments performed

| Structural Experiments | | |
|---|---|---|
| Experiment No. | Experiment Name | Description |
| SE 0 | Base Model | Establish a sufficient input buffer capacity for the cell |
| SE 1 | AGV | Use AGVs to load and unload machines. Production seven days a week and no breaks. |
| SE 2 | NoHelp | All modules are independent and do not need assistance from the robot other than for loading and unloading the modules. |
| SE 3 | ArrivalOrder | Structure the arrival order of inserts and solids to the cell. Every other tool is solid and insert. |
| SE 4 | AGV+ArrivalOrder | Combined the AGV and Arrival order experiment. |
| SE 5 | AGV+NoHelp | Combined the AGV and NoHelp experiment. |
| SE 6 | NoHelp+ArrivalOrder | Combined the NoHelp and ArrivalOrder experiment. |
| SE 7 | AGV+NoHelp+ArrivalOrder | Combined the AGV, NoHelp and ArrivalOrder experiment. |

### 4.2.6 Production runs and analysis

Before beginning the implementation of the structural and parameter improvement experiments, a base model had to be established to have a model to compare the results with. Due to lack of data, an appropriate size of the input buffer was decided through an experiment, labelled as structural experiment 0 (SE 0 in Table 4.10).

Six buffer experiments, labelled BE 1-5, were performed where the buffer size was altered from 100-200 in steps of 20.

The output from the cell increases linearly when the buffer capacity is increased linearly. However, when the buffer size is 100-140, the slope of the curve is steeper than when it is 140-200, see Figure 4.6. Thus, the importance of the buffer size with regards to the output of the cell is decreasing in value after the buffer size has reached 140. Having large buffers is usually not desirable, therefore the buffer size of 140 was determined to be a sufficient value for the base model.

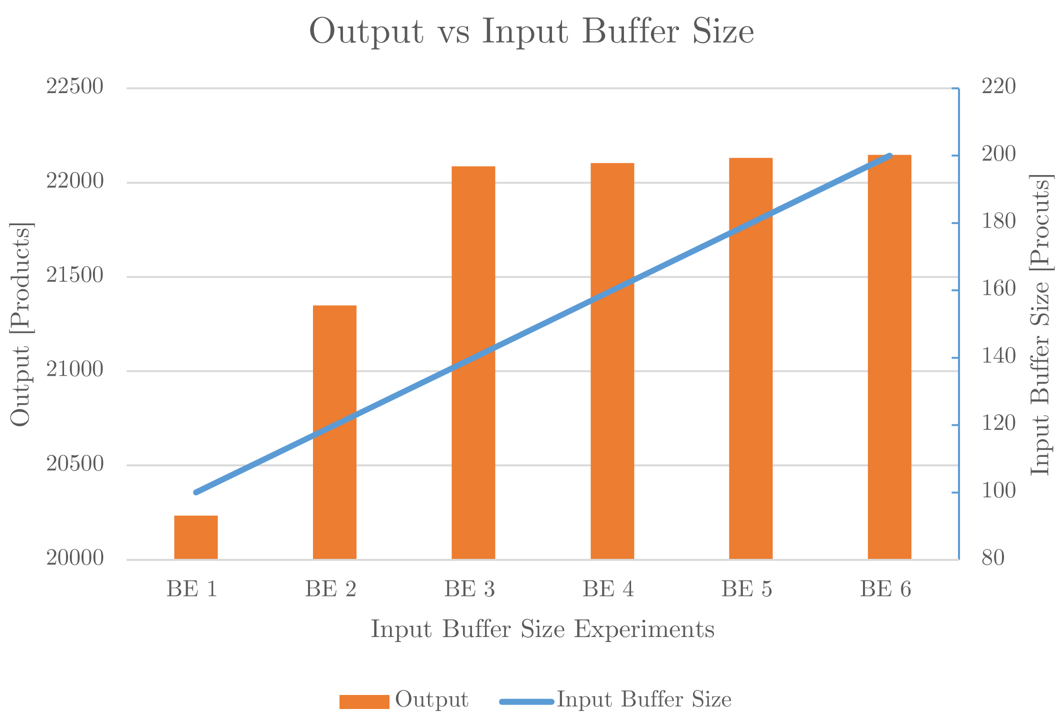



**Figure 4.6:** Input buffer size compared to the output of the cell.

Observe that when comparing the curve for the robot utilisation to the output from the cell in Figure 4.7, they are near identical in shape. The Arrow Method (see Section 3.3.3) indicates that the robot is the bottleneck of the system and its utilisation determines the cell productivity. The utilisation of the robot is between 3-4 times higher than any other machines/modules utilisation, thus it can be concluded that it is the current bottleneck of the system. The desired output of the system is roughly 73000 products, where the output of the base model is roughly 16000 products. Therefore, there is a need to increase the productivity of the flexible automation cell by 450 % to meet the current demand.

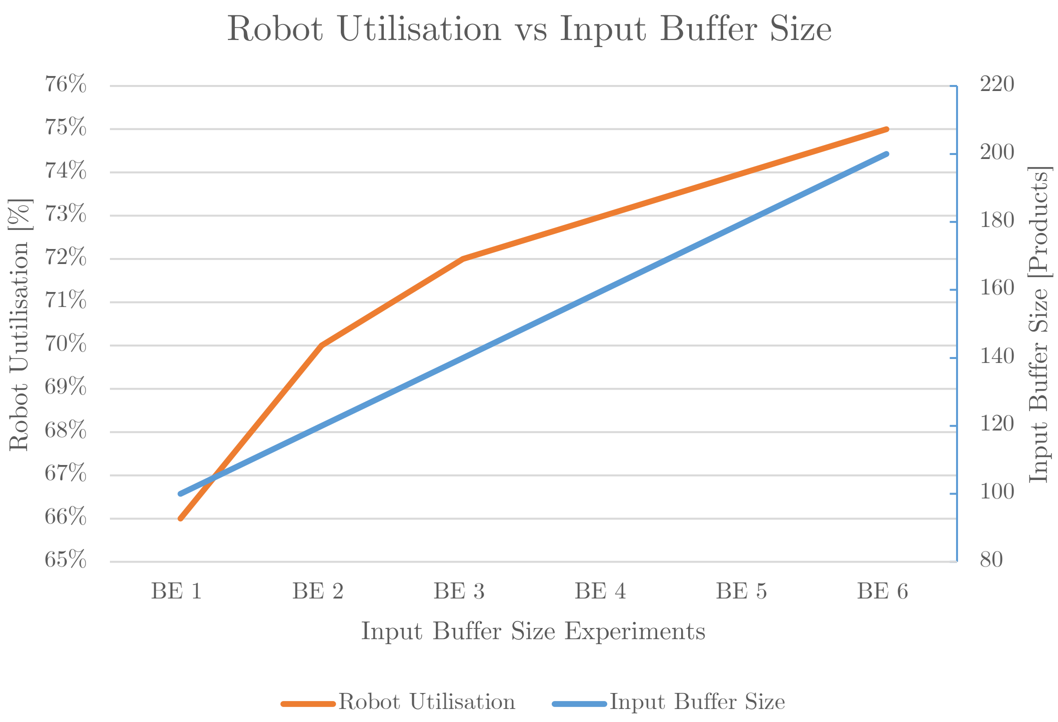



**Figure 4.7:** Input buffer size compared to the utilisation of the robot within the cell.

The structural experiments, labelled SE 0-7, were subsequently applied to the model and simulated. Results from the runs are presented in Figure 4.8. Experiment SE3, where the arrival order of the parts were controlled, did not yield any increase in the output from the cell. Neither did it have any impact when it was combined with other experiments, thus the arrival order of products is not of great importance for the capacity of the cell. SE 5, where the AGV and NoHelp experiments are combined, resulted in the highest output.

The increased output is correlated with the utilisation of the robot, as was the case with the base model. When the robot utilisation increases, the output increases for most of the experiments. The bottleneck was not broken by the experiments, but rather the strained further.

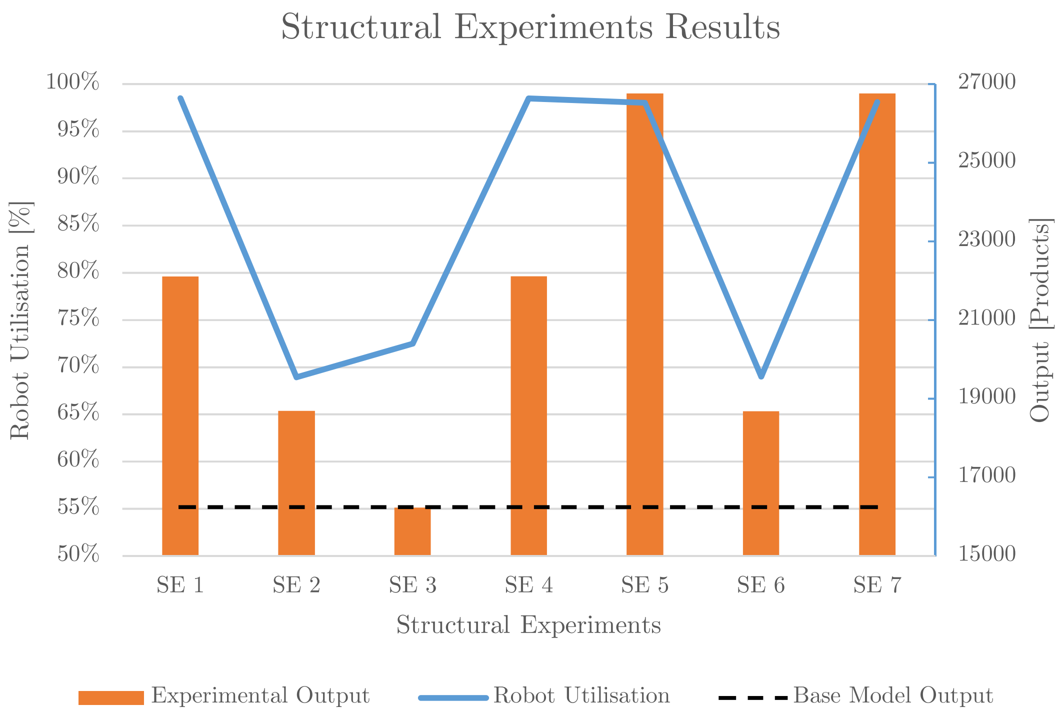



**Figure 4.8:** Experiment 1-7: Structural design changes to the model

In Figure 4.9 and Figure 4.10 one can see the effect of different parameter changes on the system and the robot utilisation. In these parameter experiments, labelled PE 1-48, the robot loading/unloading time was changed from 1 to 15 seconds in steps of 2 and the input buffer size was changed from 100 to 200 in steps of 20.

By analysing the results, one can see that the parameter that has the biggest effect on the system output is robot loading/unloading time, with the input buffer size being close to negligible. The output increases linearly with the reduction of robot loading/unloading time. The same effect can be observed concerning the robot utilisation, with the utilisation decreasing drastically when the loading/unloading time of the robot is low. This utilisation increases until the robot loading/unloading time is 7 seconds, whereby the curve flattens.

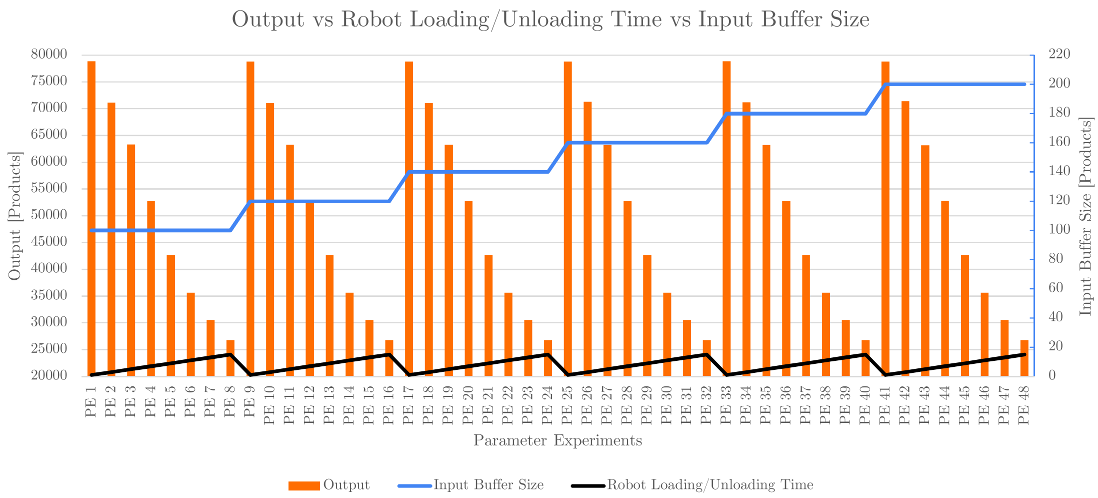



**Figure 4.9:** Output of the system when varying robot loading/unloading time and input buffer size.

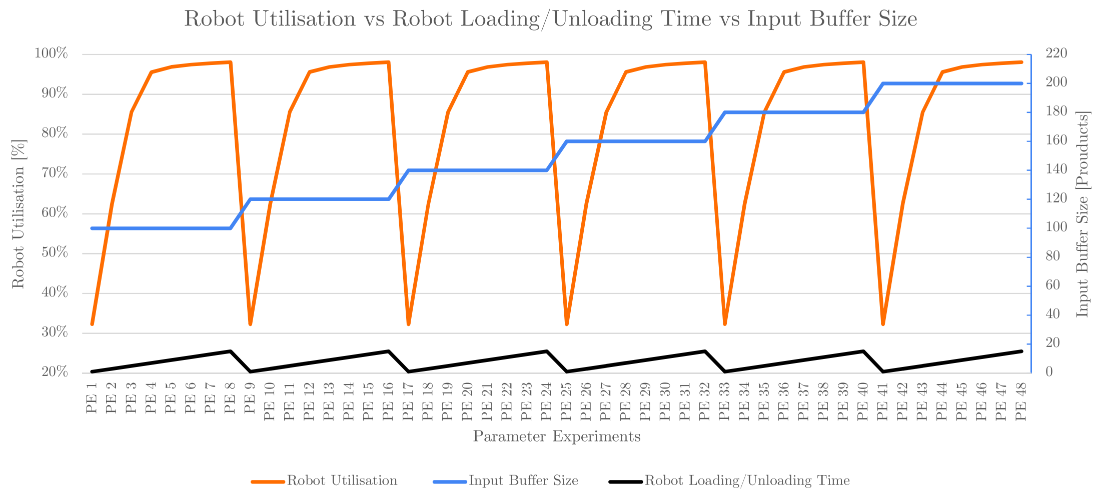



**Figure 4.10:** Utilisation of the robot when varying robot loading/unloading time and input buffer size.

### 4.2.7 Suggested system improvements

From the production runs and analysis there are several experiments that showed an increase in productivity for the flexible automation cell, providing insight into DES as a design tool for production systems. The most promising one is the implementation of AGVs to transport the tools and to load the CNC-machines, with which production can continue running without human operators. This does not take into account tasks such as maintenance and handling tools at the manual interaction module, as these tasks still have to be performed by human operators. Additionally, making the vision module and the solid tool change module independent where they do not require aid from the robot to perform its tasks, also increases the productivity of the cell.

Robot loading/unloading time is also of great importance where having a more optimised robot will benefit the productivity substantially. Currently the loading/unloading time is 15 seconds and decreasing it by 50% would result in a loading/unloading time of roughly 7 seconds. Decreasing the loading/unloading time by more than 50% could prove challenging as robot safety and accuracy is of great importance. Therefore, 7 seconds as loading/unloading time is suggested as a reasonable value.

Lastly, the input buffer size had little impact on the capacity of the cell once the AGVs were implemented. Having smaller buffers is often desirable, thus the input buffer is suggested to have a capacity of 100 tools compared to the capacity of 140 tools that the input buffer had in base model.

### 4.2.8 Sensitivity analysis

Finally, a sensitivity analysis was performed on the model with the suggested changes. The sensitivity analysis was performed since the other experiments did not take into consideration disturbances within the production, such as machine breakdowns.

The sensitivity analysis was performed by including the possibility for failures to happen in the bottleneck machine, the robot. Since there is no recorded data for how often the robot fails, the analysis was based on experimenting with multiple values of the availability parameter of the robot. The analysis was performed by having robot availability ranging from 95%-99% in five availability experiments, labelled AE 1-5, the results are presented in Figure 4.11.

The output of the cell decreased from 52714 to ranging between 43340-49904 tools replaced during 60 days of production, depending on which availability the robot has. Thus, the systems capacity was decreased by 5%-18% when including breakdowns to the robot as a disturbance.

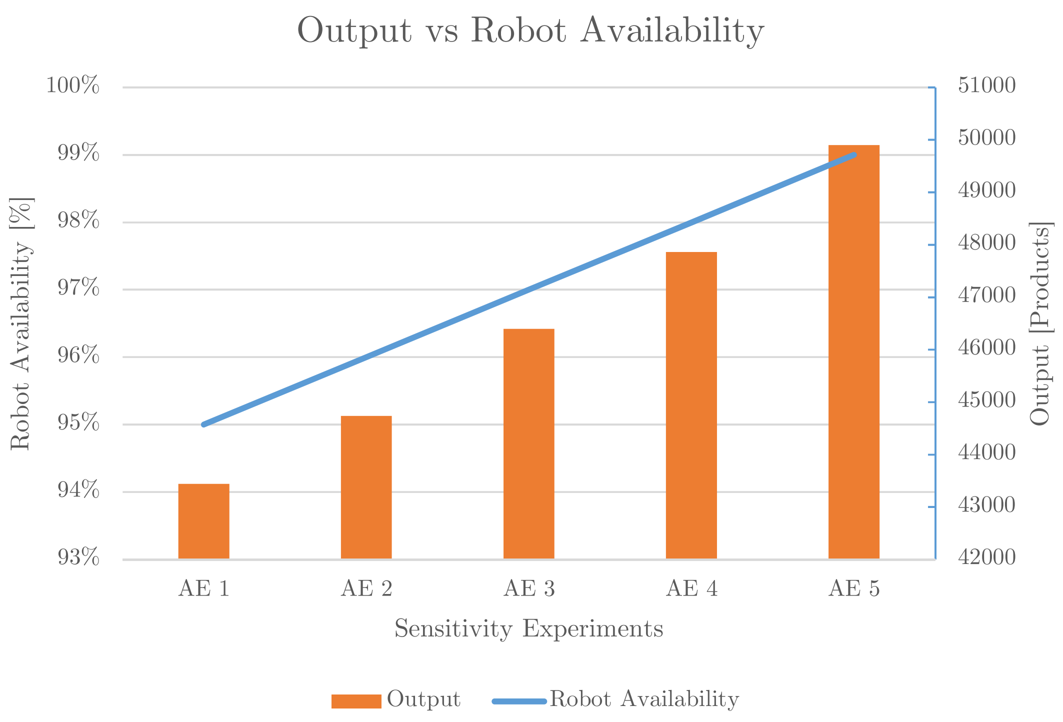



**Figure 4.11:** Availability of the robot compared to the output of the cell when the loading time is 7 seconds.

### 4.2.9 DES viability sheet results

To assist in establishing the viability of DES in this particular case, the DES viability worksheet, see Figure 3.3, introduced in Section 3.4 was filled out by the modellers. The result of this is presented in Figure 4.12. According to the scoring on the viability sheet, the studied case is in the lower half of the 51-75 score, which according to the sheet shows that the DES was suitable for the case, but with major challenges.

| DES viability worksheet based on Banks methodology | | | | |
|---|---|---|---|---|
| The higher the total score, the more viable DES was for the project | | | | |
| No. | Step | Description | Score from 1-5 | Comment |
| 1 | Problem formulation | It was simple to create a problem formulation | 3 | Predefined by stakeholders, minor adjustments made |
| 2 | Project plan | It was simple to plan the project | 3 | No previous software experience, only previous project experience |
| 3 | Project plan | The project followed the project plan closely | 4 | |
| 4 | Model conceptualisation | It was simple to create a model conceptualisation | 2 | Difficulties breaking down the behaviour of the system |
| 5 | Model conceptualisation | The conceptual model was useful when creating the operational model | 2 | The broken down behaviour was utilised, but not to an extent |
| 6 | Data collection | It was simple to collect data | 1 | |
| 7 | Data collection | The availability of data was high | 2 | A limited amount of data was available |
| 8 | Data collection | The quality of the data was high | 1 | The data quality was mostly C data |
| 9 | Model translation | The model translation was simple | 3 | Some difficulties implementing the robot behaviour |
| 10 | Model translation | The previous knowledge of the software used was high | 1 | No previous software experience |
| 11 | Model translation | The final operational model was accurately describing the real system | 3 | The completed parts of the cell were examined and compared |
| 12 | Verification | It was simple to verify the operational model | 4 | Readily available debugging tools led to easy verification |
| 13 | Verification | The operational model could be verified accurately | 4 | |
| 14 | Validation | The validation of the operational model was simple | 1 | No real-world cell capacity data available |
| 15 | Validation | The oprerational model could be validated accurately | 1 | |
| 16 | Validation | The availability of data for the validation was high | 1 | |
| 17 | Experimental design | Designing the experiments was simple | 4 | Many possible experiments provided by stakeholders |
| 18 | Experimental design | The results from the experimental designs was useful | 4 | Many suggested improvements and insights into the cells performance |
| 19 | Production runs and analysis | The production runs and analysis process was simple | 5 | Smooth process with many built-in tools in the software used for analysis and production runs |
| 20 | Production runs and analysis | The results of the analysis was valuable for the stakeholders | 3 | |
| | Extra scoring: | If the project provided improvement/design suggestions that did not come from the DES results, add 3 points per suggestions. | | |
| | | Final score: | 52 | |

| |
|---|
| 0-25 = DES was not viable for the project |
| 26-50 = DES was not suitable for the project, but provided useful results for the stakeholder. |
| 51-75 = DES was suitable for the project, but there were challenges |
| 76-100 = DES was viable for the project |

**Figure 4.12:** DES viability sheet filled out by the modellers.

## 4.3 Summary

The results of the simulation runs of the base model and the different experimental models show that the bottleneck of the system is the robot, and effort should be put into optimising the loading and unloading speed of the robot to break the bottleneck. With the application of the suggestions in SE 5 and an optimised robot speed, the daily tool demand can be met by using two robot cells for the entire production system, proving the flexible automation cell concept.

The results show that the DES model of the flexible automation cell can provide insight into how it will behave and perform when it is deployed in the real-life factory. It also shows that the model can be used as a tool to assist the development of the flexible automation cell, as the model shows that designing all the modules to not require any assistance from the robot is preferable. However, finding out how exactly these modules should be designed was not something that was possible using DES. Furthermore, different parameters such as input buffer size and robot unloading/loading times could be decided using DES, as seen in Section 4.2.5. To answer research question 1 (see Section 1.4): DES can be used to make design decisions such as buffer size and robot speed, and also provide improvement suggestions to the system, such as AGVs and completely autonomous modules. However, more complex design decisions such as the exact behaviour and logic of individual modules, was not possible using DES in the early design stages.

The experiments presented in Section 4.2.5, production runs presented in Section 4.2.6 and improvement suggestions presented in Section 4.2.7 show that DES used in the early design of production systems can provide useful insight into the future system. One can make certain design choices and improvements to the system which is currently in development, and through these improvements save time and money. This is further shown via the viability sheet results presented in Section 4.2.9, which suggests that DES was suitable for the project. To answer research question 2: The model of the flexible automation cell provided insight into the system and its behaviour, improvements to the design and the parameters of the system as well as a simulation model which can be further utilised in the future. This shows that despite a lack of data and no complete real-world system to base the model on, DES can still be viable and useful in the early design of production systems.

# 5
# Discussion

*The following chapter will provide a discussion based on the case results, the viability of DES in the early design of production systems, challenges faced using Banks, sustainability aspects of the project as well as discussion on further research.*

## 5.1 Case results

The results of the case study show that the performance of one flexible automation cell today cannot prepare enough tools to meet the daily demand. However, with the suggested improvements to the design and parameters of the system, having at least two flexible automation cells preparing tools in tandem might have the capacity to prepare enough tools.

These results cannot be confirmed to be completely accurate however, as there is no real-world system to compare the results against and the quality of the data used in the model is low. This means that the future real-world flexible automation cell might prepare more tools or less than the results suggest. The feasibility of the suggested improvement is also questionable, as knowing which improvements that can actually be implemented is difficult.

There are also aspects which have not been taken into consideration in the model, such as AGV- and module breakdowns, as these were deemed too difficult to estimate data for and implement. In the real-world, these aspects will likely affect the capacity of the flexible automation cell.

The methodology, used during the case, developed by Banks and explained in Section 2.1.1, proved to be a suitable method for the case. The method proved easy to follow and each step of the process was in a logical order. However, some steps of the process are hard to perform correctly considering production systems in an early design stage. This is mostly due to the lack of data available, making data collection and validation difficult.

## 5.2 Viability of DES in the early design of production systems

To determine the viability of DES in the early design of production systems, a case study was performed and a DES viability sheet was filled out based on the process of the case study and its results. According to these results, there is proof which indicates that DES is a viable tool to aid developers when designing new production systems and that it can provide detailed improvement suggestions. This is possible despite an overall low quality of data and no complete real-world system in which to base the simulation model on.

Even though it is impossible to create a model which captures the future real-world system with complete accuracy, it is still possible to see certain trends in the system and get a grasp of its overall performance and behaviour. These aspects can be used to make improvements to the operational model and assumptions can be made that these improvements can also be applied to the real-world system. However, in the real-world system, the improvements will be made early on and seen as design changes, rather than improvements.

Design changes and improvements that are particularly complex in nature, such as the design and behaviour of the modules, are not as viable however. To change the design and make improvements of an individual module requires a more detailed simulation which incorporates system dynamics as well as a thorough breakdown of every operation within each module process.

When evaluating DES as a design tool when comparing it to the design process presented in Section 2.3, it can be seen that DES can be useful in the evaluation of conceptual production systems and system layouts [16], [17]. In addition, DES proved useful in determining system parameters, and which parameters are interesting to monitor. This was shown in the case study, where it was found that robot loading/unloading time affects the system drastically, and that the developers should focus on reducing this time. Optimally, DES should not solely be used when designing production systems, but used in tandem with the regular design process. Using DES to evaluate parameters, layouts, flows etc. when designing production systems can prove useful in finding design errors and if these errors are not found it can lead to problems in the implementation phase [15].

The viability sheet, see Figure 4.12, shows that DES was suitable for the studied case but it was not without its challenges. However, this viability sheet is subjective, and the results can vary depending on every individual experience with a project. The questions for the steps in the viability sheet was based on common challenges when applying DES, thus they can be considered relevant. However, there are more challenges that can be included in these steps to get a more detailed evaluation of the viability. Furthermore, the evaluation of the final score where different ranges of scores was defined to deem the outcome of the project lacked a rigorous and structured method. Currently, the scoring ranges are defined by equally dividing the maximum amount of points available by the amount of outcomes available. However, this method where each outcome has equal potential should be revised since these limits were not based on any statistical evidence. The results from the evaluation of the viability sheet do however accurately reflect the views of the modellers in this case which shows the potential of this evaluation method. As the project was finished, the modellers agreed that DES provided many interesting and useful results, while at the same time having been a great challenge. These challenges are further explained in Section 5.3.

## 5.3 Challenges of DES in the early design of production systems

In the following section, discovered challenges with DES and a solution to these challenges will be presented. They will also be compared to the common challenges found by Florecs-Garca *et al.*[23].

### 5.3.1 Model conceptualisation

According to Florecs-Garca *et al.*[23], the model conceptualisation and making an adequate system abstraction is one of the major challenges in a DES project in the early design stages of a production system. This proved to be the case in this project as well, confirming it as a major challenge. As there is no finished system to analyse and make an abstraction out of, it is difficult to make sure that the conceptual model actually represents the real system accurately. Therefore, it is of great importance to discuss with the stakeholders and experts of the system how the real system is supposed to behave and what simplifications can be done. The approach of converting a flow chart created by the system experts of the future system into SFCs proved to be sufficient in this case, however, it is still a substantial challenge to accurately break down the logical behaviour of the system for it to be easily translated into an operational model. This is especially the case when the system needs to be modelled with a high level of detail, wherein DES might not be the most suitable solution. Instead, a hybrid simulation approach as mentioned in Chapter 2.1 might be an alternative.

### 5.3.2 Data Collection

The data collection phase proved to be one of the more challenging tasks of the DES modelling project. This confirms that data collection in early design stages of production systems is a major challenge, as was also described by Florecs-Garca *et al.*[23].

The flexible automation cell is not yet finished and some of its modules are not physically constructed yet. Thus, the input data had to be based on an mixture of data: how the production currently performs, such as CNC-machine processing time and manual tool preparation, and how the robot and the modules that are actually constructed performs. Having a mixture of data where there are different owners of the data is something that delayed the data gathering process, as the data had to be gathered by first finding the relevant data owners and contacting them with the assumption that there is actually data available, which was not always the case.

Furthermore, approximating certain data parameters that are based on how the production system currently performs can not be considered to be an accurate method. It is hard to predict exactly how the flexible automation cell will perform some of the processes, and if it will perform better or worse when considering the time spent on each process.

It is also hard to define the accuracy of the model since there is nothing to compare the model to. Most of the input data is of classification C, thus it can be assumed that it is of rather low quality and not high accuracy [13]. However, since this model is to be used as a design tool to try different concepts and how they affect the flexible automation cell, the accuracy might be sufficient. The analysis performed showed trends and behaviour within the flexible automation cell that can be used to assist the system design, but it does not have enough accuracy and detail to define operations that requires higher accuracy such as daily planning of the production.

To deal with the discussed challenges, one has to decide on relevant parameters early on in the modelling process. The reason for this is because data collection takes time, and having decided what data is needed early on means that the relevant parties can be contacted as early as possible. It is challenging to know exactly what parameters are needed early on in the modelling process, and some parameters are found when collecting other data. Therefore, having the data collection process started early on with the relevant parameters required, leads to interesting results in a lower amount of time.

### 5.3.3 Model Translation

The limitations of the model conceptualisations stage substantially affected the later model translation stage. Uncertainties and how to deal with them as well as choosing a relevant abstraction level that the system should be modelled by impacted how the model translation was performed. Having a rather unspecific model conceptualisation and trying to build the model in the software without knowing how some of the parts of the model should be constructed and how the real system should behave caused the translation stage of the project to require more time than expected.

Whilst the implementation was performed, additional information about the real-life system was found which impacted how the system was modelled. This lead to the modellers having to make rather extensive changes to the code and change the approach entirely to satisfy the features of the real-life system which was not included in the model conceptualisation. This proved to be one of the contributing factors to the increased time consumption in this stage. Furthermore, the modellers inexperience with the modelling software made the model translation stage more difficult, as the modellers did not know what tools that could be used. Not knowing what the capabilities of the software was and how it could be used to model the system caused the model development to be based on a trial and error type of structure. Each area of the conceptual model was coded independently and tested without knowing how these independent areas should be linked together to form the model. Some of the tools that were used within the software to create these independent functionalities and link them together proved to be unsuitable. At that point, a considerable amount of time had already been spent developing the operational model with these tools. Having to change the structure of the operational model at that point caused the overall model translation stage to be delayed further.

To prevent or to minimise the obstacles that were faced in the model translation stage there are mainly two things that could have been done. Firstly, the model conceptualisation should be done with a more thorough approach. The stakeholders should be included early on to make sure that the behaviour of the system is understood and what the model should be used for to make sure that the conceptual model and the model translation is done correctly without having to make unforeseen changes. Lastly, to prevent poor decision making when deciding what tools to use within the software and how to structure the conceptual model, the learning phase should be more extensive. Modelling a complex system without having sufficient knowledge of the chosen software proved to be very time consuming. By having a more extensive learning phase where experience and knowledge about the software is gathered is crucial to make better design decisions that has a higher probability of making the conceptual model run as intended.

### 5.3.4 Verification and validation

The verification process consisted of thoroughly debugging and testing the code in an effort to ensure that it behaved to the established conceptual model and specifications. It was a very arduous and time consuming process, as when one issue was resolved, another one appeared. During this process, the fact that the modellers had limited prior knowledge of the software became clear. The modellers did not become aware of many efficient ways of debugging until later in the project, and as such a considerable amount of time spent debugging could have been reduced had the knowledge of the software been sufficient. Therefore it is important to have prior knowledge of the software, or a longer learning phase before the actual modelling begins.

As there is physically no complete existing system upon which the model is based, validation of the model was evidently challenging. There is no data on how the system is supposed to perform, only the demand. Therefore it was challenging to know whether the system performed as intended or not. In an effort to validate the system, the system developers and experts were consulted, where they examined the model to see if it behaved as they had originally intended. The experts deemed that the model behaved as expected, and the model was deemed semi-validated.

## 5.4 Sustainability aspects

Concerning sustainability aspects, the different ways the flexible automation cell will affect the **social**, **environmental** and **economic** pillars of sustainability has to be considered [25]. The **social** pillar is about the people connected to the company, for example the employees and the stakeholders. To be socially sustainable, the company has to make sure that the employees are treated fairly, that their workplace is safe and everyone in the supply chain is taken into consideration [25]. The **environmental** pillar is about how the company affects the environment. To be sustainable the company has to focus on, for example, the reduction of waste, decreasing its $CO_2$ emission and water usage [25]. The **economic** pillar is about the profitability of the company, as to be sustainable, the company has to be profitable. However, profitability can never be more important than the **environmental** and **social** pillars [25].

As the flexible automation cell is designed to automate the tool preparation process which today is done manually by operators, it is only natural to assume that these operators will no longer be required to perform this specific process. This could potentially lead to layoffs within the company. However, there are still operators required to operate the flexible automation cell, and as such the operators tasks would change from manually preparing the tools to operating the flexible automation cell. However, the flexible automation cell would probably require fewer operators than the manual preparation of tools does.

The act of automating the tool preparation would in general decrease the ergonomic strain upon the operators. They would no longer be required to perform the physically and cognitively repetitive task of preparing the tools manually. Repetitive tasks are considered by Berlin and Adams [22] to be one of the most harmful factors. This is especially the case when considering tasks which are performed mostly with the fingers and hands [22], which is the case of the manual preparation of tools. The monotonous and repetitive nature of the task may also lead to the operators experiencing boredom. This might cause the operator to make errors when performing the task, as the mind of the operator slips [22]. The ergonomic issue of the tool preparation process would therefore be effectively eliminated with the introduction of the flexible automation cell within the company's manufacturing system.

When considering the **environmental** pillar of sustainability, the biggest aspect to consider is the fact that the flexible automation cell and its components has to be produced, which affects the environment. The production of the automation cell would likely cause a certain amount of $CO_2$ emissions and water usage. This will also require energy and if the energy is not renewable, such as wind power and solar power [26], it will have a negative effect on the environment. Energy will also be required when the automation cell is in operation, which is not required in the current situation where the tools are prepared manually.

The last pillar to be considered is the **economic** pillar of sustainability. With the introduction of the flexible automation cell within the manufacturing system, tools for the machines could be prepared automatically by the machine at night, when no employees are working. This could potentially lead to an increase in productivity of the entire production system, leading to profit. The fact that the discrete event model of the flexible automation cell can be utilised to find improvements to the cell before it has been finished can be seen as profitable as well. This is because one does not have to rebuild the cell to make adjustments, instead the cell can be improved in earlier stages.

## 5.5 Future research

To further evaluate the potential of DES as design tool in early design stages of production systems, research should be done on a hybrid of DES and continuous simulation. By utilising continuous simulation as well as DES, system dynamics and complex system behaviour can be modelled while at the same time taking advantage of the potential of DES. With a hybrid approach, one can experiment and find improvements to the entire production system and its parameters, but also find design improvements to the more complex parts of a production system. In this case, the complex parts are the different modules of the flexible automation cell, which all have complex individual behaviour.

The method of how to evaluate whether or not DES is viable or not that was used in this project has areas that can be improved further. Defining the scoring ranges that are used when evaluating the results from the viability sheet or defining a new evaluation method is something that should be investigated further.

Additionally, the viability analysis performed in this case study was based on the results from the application of DES. Research to create a viability analysis that can be performed to assess whether or not DES should be used as a tool would be helpful, prior to the launch of a project. This has a potential of assisting company's in choosing suitable tools that are beneficial for their application and project.

## 5.6 The covid-19 pandemic

The covid-19 pandemic affected the data collection by removing the possibility of collecting manual tool preparation data from the physical factory. To solve this problem, a tool preparation expert was brought from the factory and observed and timed when performing the manual tool preparation tasks. Because there was a limited amount of operators to observe, the data is not of the highest quality, reducing the overall accuracy of the model. This could have been resolved by collecting data earlier in the process, showing that data collection should start as early as possible as discussed earlier in Section 5.3.2.

Furthermore, covid-19 also prevented the modellers from going to the company's location. To solve this problem, the modellers received a computer with a Plant

Simulation license in order to work from home and all meetings with the supervisors were relegated to online meeting. This proved a suitable solution, and with the exception of data collection, the thesis was minimally affected by the covid-19 pandemic.

Despite covid-19 complicating parts of the process, the pandemic showed that a DES study does not have to be conducted at the physical factory. If no data has to be collected, or the data can be collected by someone working at the factory, a DES study can be performed completely from home. All necessary meetings can be held online, and this has the potential to save time for the modellers. Travelling to and from the factory takes time, and not being able to travel meant that this time was instead spent on the DES study. This has the potential of reducing the overall time a DES study takes, saving the company money, and not having to travel also reduces the environmental impact. Despite a DES study not affecting the environment heavily or at all, travelling to and from the factory does. The pandemic has made people realise that physically being at work is not a requirement, and working from home can be as effective.

# 6
# Conclusion

This thesis has shown that DES can be a viable tool to assist the development of future production systems and production cells. It was possible to achieve results showing how the future system might perform with the use of estimated input data. Analysing these results provided information regarding the main bottleneck of the system, which proved to be the robot of the flexible automation cell. Furthermore, the results from the experimental design and production runs showed an estimation of the capacity of the cell, and that the performance of the robot greatly influenced the cells capacity. Implementing the suggested improvements and focusing on optimising the performance of the robot has the possibility to make flexible automation cell viable in daily production.

This shows that DES utilised in the early design stages of a production system is a viable approach. DES can influence the design of the future system, and be used as a tool to decide on production parameters. This in combination with the DES viability sheet provides proof that DES is viable in the early design of production systems and that its ability to experiment with design suggestions and parameters is powerful. Although limited in its ability to provide complex design improvements on system behaviour and logic, its advantages outweigh its disadvantages, and is a useful venture for manufacturing companies looking to design new production systems or experiment with existing ones.

# Bibliography


[1] E. M. Goldratt and J. Cox, *The goal: a process of ongoing improvement.* Routledge, 2016.

[2] *Gkn aerospace in sweden*, `https://www.gknaerospace.com/en/about-gkn-aerospace/locations/gkn-aerospace-in-europe/gkn-aerospace-in-sweden/`, Accessed: 2020-01-28.

[3] R. Nnaji, M. Bodude, L. Osoba, O. Fayomi, and F. Ochulor, "Study on high-temperature oxidation kinetics of haynes 282 and inconel 718 nickel-based superalloys", *The International Journal of Advanced Manufacturing Technology*, pp. 1–12, 2019.

[4] S. J. Eric and A. Seco Tools, *Metal cutting theories and models.* Division of Production and Materials Engineering, 2012.

[5] G. S. Fishman, *Discrete-event simulation: modeling, programming, and analysis.* Springer Science & Business Media, 2013.

[6] J. Banks, *Discrete-event system simulation.* Pearson Education, 2010, ISBN: 9780138150372.

[7] S. Bangsow, *Tecnomatix Plant Simulation.* Springer, 2015.

[8] T. Sobottka, F. Kamhuber, M. Rössler, and W. Sihn, "Hybrid simulation-based optimization of discrete parts manufacturing to increase energy efficiency and productivity", *Procedia Manufacturing*, vol. 21, pp. 413–420, 2018.

[9] M. Armendia, M. Ghassempouri, E. Ozturk, and F. Peysson, *Twin-Control: A Digital Twin Approach to Improve Machine Tools Lifecycle.* Springer, 2019.

[10] J. Bokrantz, A. Skoogh, D. Lämkull, A. Hanna, and T. Perera, "Data quality problems in discrete event simulation of manufacturing operations", *Simulation*, vol. 94, no. 11, pp. 1009–1025, 2018, ISSN: 00375497. [Online]. Available: `http://search.ebscohost.com/login.aspx?direct=true&AuthType=sso&db=asx&AN=132458864&site=eds-live&scope=site&custid=s3911979&authtype=sso&group=main&profile=eds`.

[11] M. Allen, A. Spencer, A. Gibson, J. Matthews, A. Allwood, S. Prosser, and M. Pitt, "Right cot, right place, right time: Improving the design and organisation of neonatal care networks  a computer simulation study", *Health Services and Delivery Research*, vol. 3, pp. 1–128, May 2015. DOI: `10.3310/hsdr03200`.

[12] A. Skoogh and B. Johansson, "A methodology for input data management in discrete event simulation projects", Dec. 2008, pp. 1727–1735. DOI: `10.1109/WSC.2008.4736259`.

[13] S. Robinson and V. Bhatia, "Secrets of successful simulation projects", Jan. 1996, pp. 61–67, ISBN: 0-78033018-8. DOI: `10.1145/224401.224424`.

[14] N. T. Thomopoulos, “Statistical distributions”, *Cham, Switzerland: Springer*, pp. 77–84, 2017.

[15] C. Rösiö and J. Bruch, “Focusing early phases in production system design”, in *IFIP International Conference on Advances in Production Management Systems*, Springer, 2014, pp. 100–107.

[16] H. B. Maynard and K. B. Zandin, *Maynard’s industrial engineering handbook*, Sirsi) i9780070411029. 2001.

[17] M. Bellgran and E. K. Säfsten, *Production development: design and operation of production systems*. Springer Science & Business Media, 2009.

[18] R. Jones, *Keyword Intelligence: Keyword Research for Search, Social, and Beyond*, 1st. USA: SYBEX Inc., 2011, ISBN: 1118061837.

[19] M. Gesvret and P. Foltin, *Conceptual modeling for discrete simulation of supply chain*, 2019. [Online]. Available: `http://proxy.lib.chalmers.se/login?url=https://search.proquest.com/docview/2311896488?accountid=10041`.

[20] N. E. Battikha. ISA, 2018, ISBN: 978-1-945541-38-4. [Online]. Available: `https://app.knovel.com/hotlink/toc/id:kpCHMCE011/condensed-handbook-measurement/condensed-handbook-measurement`.

[21] L. Li, Q. Chang, and J. Ni, “Data driven bottleneck detection of manufacturing systems”, *International Journal of Production Research*, vol. 47, no. 18, pp. 5019–5036, 2009.

[22] C. Berlin and C. Adams, “Production ergonomics”, 2017.

[23] E. Flores-Garca, J. Bruch, M. Wiktorsson, and M. Jackson, “Challenges of discrete event simulation in the early stages of production system design”, *International journal of industrial engineering*, vol. 26, no. 5, pp. 819–834, 2019.

[24] F. Freiberg and P. Scholz, “Evaluation of investment in modern manufacturing equipment using discrete event simulation”, *Procedia Economics and Finance*, vol. 34, pp. 217–224, 2015.

[25] A. Beattie, *The 3 pillars of corporate sustainability*, `https://www.investopedia.com/articles/investing/100515/three-pillars-corporate-sustainability.asp`, 2019.

[26] M. Kaltschmitt, W. Streicher, and A. Wiese, *Renewable energy: technology, economics and environment*. Springer Science & Business Media, 2007.

**DEPARTMENT OF INDUSTRIAL & MATERIALS SCIENCE**
**CHALMERS UNIVERSITY OF TECHNOLOGY**
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY