A Study of Reconfigurable Manufacturing Systems with Computer Simulation

Jacques du Preez

Thesis presented in partial fulfilment of the requirements
for the degree of Master of Science in the Faculty of
Industrial Engineering at Stellenbosch University

Study Leader: Mr. J. Bekker

Date: December 2011

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 20 November 2011

Abstract

Reconfigurable Manufacturing Systems (RMSs) have the ability to reconfigure hardware and control resources at all of the functional and organizational levels. This allows for quick adjustment of production capacity and functionality in response to sudden changes in market or in regulatory requirements.

This study evaluates the characteristics and operation of automated reconfigurable assembly lines using discrete event simulation. The assembly line uses a conveyor system which transports pallets to various machines to perform the assembly process. Different conveyor configurations are developed for the same assembly process using Simio simulation software. A part family consisting of five variants are assembled on the same assembly line with a large variation in the production quantities for each product. This requires the assembly system to be able to quickly adjust its functionality and capacity.

Multi-objective optimization is performed on the models through the use of a Pareto exhaustive search experiment. The two contradicting objectives used are the *throughput rate* of the system and the average *work in progress*, with the aim of maximizing the former and minimizing the latter. From the Pareto exhaustive search experiment, a Pareto front is constructed showing which configuration is preferred under certain operation conditions. However it is concluded that the Pareto front can be tailored to fit the specific needs of the decision maker, depending on what the decision maker is willing to pay.

An experiment that evaluates the effect of changing the conveyor speed is performed. It is established that under certain operating conditions, increasing the conveyor speed higher than the ceiling value will not improve the performance of the system.

A production scenario was also developed which include different order sizes for each of the five parts of the part family. The configurations have to alter their capacities based on the order sizes to test which system performs the best under these operating conditions. For this experiment, the ramp-up time was of interest but the best system was chosen based on the combination of throughput rate and the average work in progress.

From the results of the different experiments, it is recommended to first determine the maximum capacity and the operating logic before choosing one of the configurations. Once this is decided, the information gathered from the experiments can then be tailored for the decision maker to establish the best operating conditions for the chosen configuration. The developed simulation models are used as a Decision Support System for future research on the topic. It is recommended for future research to focus on using Automated Guided Vehicles (AGVs) instead of a conveyor system as transportation method.

Opsomming

Herkonfigureerbare Vervaardigingstelsels (HVSs) het die vermoë om alle hardeware en beheer hulpbronne, op alle funksionele en organisatoriese vlakke te herkonfigureer. Dit maak dit moontlik vir vinnige verstellings aan produksie kapasiteit en funksionaliteit, indien daar 'n skielike verandering in die mark of wetgewing is.

Hierdie studie evalueer die karakteristieke en werking van outomatiese herkonfigureerbare monteerlyne met behulp van diskrete gebeurtenis simulasie. Die monteerlyne gebruik vervoerbande om pallette na verskeie masjiene te vervoer, sodat die parte aanmekaar gesit kan word. Simio simulasie sagteware is gebruik om verskillende vervoerband konfigurasies vir dieselfde monteringsproses te ontwikkel. 'n Part familie van vyf variante word op dieselfde monteerlyn aanmekaargesit. Daar is 'n groot variasie in die produksie hoeveelhede van elk van die vyf variante, dus moet die monteerlyne vinnig die kapasiteit en funksionaliteit kan aanpas.

Multi-doelwitoptimering is toegepas op die modelle deur 'n Pareto alomvattende soek eksperiment uit te voer. Die twee teenstrydige doelwitte wat gebruik is, is die *deurset tempo* van die stelsel asook die *gemiddelde werk-in-proses*. Die doel is om die deurset tempo te maksimeer en terselfde tyd die gemiddelde werk-in-proses te minimeer. Die Pareto alomvattende soek eksperiment word verder gebruik om 'n Pareto front te skep wat uitwys watter vervoerband konfigurasies verkies word onder sekere bedryfstoestande. Die Pareto front kan egter aangepas word om die spesifieke behoeftes van die besluitnemer te pas.

'n Eksperiment is uitgevoer om die uitwerking van die vervoerbandspoed op die stelsel te toets. Resultate het getoon dat onder sekere bedryfstoestande die werkverrigting van die stelsel nie verbeter indien die spoed 'n maksimum grenswaarde oorskry nie.

'n Eksperiment wat 'n produksie scenario voorstel is ontwikkel waarin die vraag na die vyf part variante gevarieer word. Die vervoerband konfigurasies moet dan die kapasiteit aanpas gebaseer op die vraag na die parte. Die doel van die eksperiment is om te toets watter konfigurasie die beste vaar onder hierdie bedryfstoestande. Die tyd wat dit neem vir die stelsel om weer op dreef te kom na 'n verandering in kapasiteit is ondersoek in hierdie eksperiment, maar die beste stelsel is nog steeds gekies gebaseer op die kombinasie van deurset tempo en die gemiddelde werk-in-proses.

Gegewe die resultate van die verskillende eksperimente, word dit voorgestel dat die besluitnemer eers die maksimum kapasiteit en die bedryfstoestande vasstel, voordat 'n vervoerband konfigurasie gekies word. Sodra dit besluit is, kan die inligting wat tydens die eksperimente ingesamel is, aangepas word om die beste bedryfstoestande vir die konfigurasie wat gekies is, vas te stel. Die simulasie modelle wat ontwikkel is word gebruik as 'n besluitnemingsondersteuningstelsel vir toekomstige navorsing oor die onderwerp. Dit word voorgestel dat toekomstige navorsing die moontlikheid van geoutomatiseerde begeleide voertuie (GBV), in plaas van vervoerbande as vervoermiddel, ondersoek.

Acknowledgements

I wish to express my sincere gratitude to everyone who has contributed to this thesis in any way. In particular, I would like to thank the persons below:

- Mr. J. Bekker for his valuable advice, constructive criticisms, guidance, and a great sense of humor.
- Professor A.H. Basson for making the funds available to complete this research.
- My parents for giving me the opportunity to study and supporting me throughout this thesis.
- Michelle for the coffee breaks, encouraging words and just being an awesome girlfriend.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER *1*

INTRODUCTION

Since the beginning of time humans have been searching for ways to improve their quality of life. Over time, this need to improve, has fueled the creation of the world we know today. Almost everything that humans use to survive in the modern world must be manufactured. In order to meet the needs of the global population, major industries were created and are still expanding.

Industries can be subdivided into two general categories: Services and Manufacturing. The service industries do not deliver a tangible product but rather a service that people cannot perform on their own. Manufacturing industries do make tangible products that can be sold at a competitive price.

According to Groover (2008) a manufacturing system can be defined as a collection of people, equipment and procedures organized to perform the manufacturing operations of a company. Manufacturing systems can also be subdivided into two other categories:

1. *Facilities*: The facilities consist of the factory, machines, tools and the equipment required to perform manufacturing activities.

2. *Manufacturing Support Systems*: It is a set of procedures to manage production and solve the technical and logistics problems encountered by the manufacturing company. The support system also manages Work in Progress (WIP), quality inspections and production planning. Certain business functions can also be included in manufacturing support systems.

As technology changes the needs of consumers also change. Today, more varieties of products are made available for consumers, while the life cycles of the products get shorter and shorter. This accelerating trend requires the redesign and replanning of manufacturing systems more frequently using shorter lead times. The challenge is that manufacturing systems must be able to be reconfigured on demand within a short amount of time. Flexibility, responsiveness and reconfigurability are key requirements to survive in the modern market (Tang & Qiu, 2004). Both the facilities and manufacturing support systems need to meet these requirements.

In order to develop and evaluate a system that meets the requirements of the modern market, an overview of the literature is given in chapter 2. Different manufacturing systems that gave rise to the concepts of a Reconfigurable Manufacturing System (RMS) are discussed. The type of parts that are produced and the machinery required are presented. The concepts of reconfiguration are explained to create an understanding how it affects the system. The analysis techniques used in the remainder of the project are also introduced and explained. Chapter 2 ends with the research problem description.

Chapter 3 will focus on the models that were developed to evaluate the characteristics and operation of a RMS. The models that were developed are used in experiments to generate output data. Chapter 4 presents the different experiments that were conducted. The controls and different input data used during each experiment are discussed in detail. The output or results generated by the experiments are analyzed and discussed in chapter 5. Finally in chapter 6, conclusions regarding the results of the study and recommendations for future research are made.

CHAPTER *2*

OVERVIEW OF LITERATURE

In the previous chapter the purpose and need for manufacturing systems were stated and a definition of a manufacturing system was given. In this chapter three types of manufacturing systems are discussed which include dedicated, flexible, and reconfigurable manufacturing systems. Dedicated and flexible manufacturing systems are well known and have proven their worth in the market. Like all systems, they have their advantages but unfortunately also some drawbacks. Reconfigurable manufacturing is a rather new concept which uses the characteristics of both dedicated and flexible systems to form a different manufacturing system.

Part families are discussed to highlight the advantages of using a reconfigurable manufacturing system. The type of machinery that is required for a reconfigurable manufacturing system is explained before the process of system reconfiguration is discussed. System reconfiguration results in ramp-up times that must be explained.

An overview of simulation as an analysis technique is given as well as the basic modeling concepts that are used throughout this research project. Simulation studies generate output data that must be analyzed with the statistical methods that are discussed in section 2.8.

Multi-objective optimization is discussed briefly since its concepts are used during the study. The chapter concludes with the problem description and the aim of this research project.

2.1 Manufacturing Systems

As stated previously, a manufacturing system consists of a collection of people, equipment and procedures (Groover, 2008). This can be seen as the building blocks required to create a manufacturing system. Manufacturing systems differentiate themselves by using different people, equipment and procedures to manufacture products. Organizing the same people and equipment in different ways, also lead to completely different manufacturing systems. The ever changing market forces companies to continuously reorganize the way they manufacture products.

Manufacturing systems have evolved from traditional to conventional to advanced systems. Examples of these systems are given to explain how reconfigurable manufacturing systems came to be. Each type of manufacturing system has its own advantages, disadvantages and specific applications depending on market demand.

2.1.1 Dedicated Manufacturing System (DMS)

Abdi & Labib (2003) describe a Dedicated Manufacturing System (DMS) as a traditional method of manufacturing. DMSs were designed to have a fixed process technology which means that the process was designed to produce a single product in high volumes. It was not designed to adapt to significant product variation. A simple example of a DMS is a brewery with a bottling plant. The brewery and bottling plant are able to produce high volumes of beer, but are not equipped to start producing soft drinks. A whole new dedicated manufacturing line will have to be constructed in order to start producing soft drinks.

The factory floor usually consists of several transfer lines that are based on inexpensive fixed automation. Each dedicated line will produce a single part at a high production rate. The high production rate is achieved by the simultaneous operation of several tools (Koren *et al.*, 1999). The parts are then transferred to an assembly line where they are assembled into the final product. From the example, the beer (a part) is produced separately and assembled (bottled) on an assembly line. Quality inspection is done at various stages in the manufacturing process.

Producing a single product in high volumes creates economies of scale. The principle of economies of scale is that the unit cost per product is reduced if it is produced in large quantities. For high volume production to be economically viable, and utilize the benefits of economies of scale, it is required that the demand for the products must exceed the supply, so that the system can operate at maximum capacity. With increasing pressure from global competition and overcapacity built worldwide, it is rare that a production line can operate at full capacity all of the time (Koren *et al.*, 1999).

It is clear that the objective of DMSs are to produce specific products at high speed and high volumes (Mehrabi *et al.*, 2000). The advantages of this method are that the products can be produced at low cost and multi-tool operation is possible (Koren *et al.*, 1999). Many companies are successful using a DMS, but these companies are usually large and dominate a specific market niche. For smaller companies, DMSs are difficult to implement due to the high initial capital cost that is required to start a manufacturing line of this nature. For smaller companies that must be able to adapt their product, to meet market demand, the main drawback of a DMS is that the system is not flexible. It can only produce one type of product. Another drawback is that a DMS is designed to have a fixed capacity. This means that the capacity cannot be scaled easily and it also becomes expensive if the system does not operate at full capacity (Koren *et al.*, 1999).

2.1.2 Flexible Manufacturing System (FMS)

Abdi & Labib (2003) describe a Flexible Manufacturing System (FMS) as the conventional approach to manufacturing. DMSs are useful as long as the market demand is high and product variation is low. In some industries the market demand varies tremendously because consumer preferences change over time, which reduces the life cycle of a product. There are also many competitors in the market that try to differentiate themselves in order to sell more products. Competitors can produce essentially the same product but with added features or functionality to create a new model of an existing base product. These small variations in product composition have large impacts on the methods of manufacturing.

Companies are forced to produce a larger variety of products to stay competitive in the market. In order to produce a larger variety, the parts of the products are grouped into part families. Part families form an important part of the manufacturing system and will be discussed in detail in a subsequent section. To produce these part families, a flexible manufacturing system is required. New technologies have made it possible to create a flexible manufacturing system that consists of expensive, general-purpose computer numerically controlled (CNC) machines and other programmable automation (Koren *et al.*, 1999).

CNC machines are designed to be general-purpose machines, which means that they can manufacture a large variety of parts that can be used in many different products. These machines are useful in make-to-order scenarios where the company manufactures custom made products for each order. Hence these machines are not designed to manufacture specific parts for the company but rather a wide variety of parts. A job shop that manufactures once-off products for an order is an example of a FMS. Another example of a FMS is when the part being manufactured requires a complex machining sequence. Typical milling and lathe machines may not be able to manufacture the part which forces the company to use a CNC machine.

CNC machines are predominantly used in FMSs due to their flexibility, high functionality and ability to produce part families, but unfortunately the high functionality of the machines causes it to be expensive. The main drawback of CNC machines is that most companies do not need all the functions that a CNC machine can provide. Hence the user pays for functionality that is not used which results in lost capital for the company.

The economic objective of a FMS is to make it possible to manufacture several types of products, that change over time, with shortened changeover time, on the same system at the required volume and quality in a cost-effective manner (Mehrabi *et al.*, 2000). In order to reach this objective, a large capital outlay is required which can be a problem for smaller companies. Once a FMS is implemented, its throughput is low compared to DMSs because FMSs use single tool machines while DMSs use dedicated manufacturing lines (Koren *et al.*, 1999). However, FMSs are more flexible than DMSs because they can produce part families that can be used in several products instead of only single parts used in one product. Another advantage of a FMS is that the capacity of the system can be increased or decreased almost instantly. This is referred to as scalable capacity and it is achieved by adding or removing CNC machines from the system. CNC machines can work in parallel to produce the same part, or a different part, at the same time due to their high functionality. Having this parallel manufacturing line simplifies the scaling of system capacity.

The concept of FMSs is a step in the right direction for producing part families but there are still too many drawbacks of which Mehrabi *et al.* (2000) identified a few:

1. It is *expensive*, since in many cases the system includes more functions than needed.
2. *Inadequate system software* is utilized, since developing user-specific software is expensive.
3. FMS are *not highly reliable*.
4. FMS are subject to *obsolescence* due to advances in technology.

In sections 2.1.1 and 2.1.2 DMSs and FMSs were discussed and they highlighted the applications, advantages and disadvantages of each type of system respectively. FMSs and DMSs were compared and it can be concluded that the flexibility and functionality of FMSs are required but also the simplicity, reliability and high throughput of DMSs. A new approach to manufacturing is required that can use the advantages of DMSs and FMSs, with as little drawbacks as possible. This need for a new type of manufacturing system gave rise to the development of a Reconfigurable Manufacturing System (RMS).

2.1.3 Reconfigurable Manufacturing System (RMS)

Abdi & Labib (2003) describe a Reconfigurable Manufacturing System as one of the advanced methods of manufacturing. In sections 2.1.1 and 2.1.2 the concepts of DMSs and FMSs were discussed. Each of these systems have their own advantages, disadvantages and applications. The market environment for smaller companies has changed since the introduction of DMSs. FMSs have tried to adapt to the market with little success mainly due to the fact that it is an expensive solution.

Bi *et al.* (2008) have identified the critical requirements for a manufacturing system so be successful in the modern market. They are:

1. *Short lead-time*: A short product lead-time affects the performance of the company in a number of ways. Firstly, if a product is introduced first or early, there is an advantage over competitors since they take longer to match the new product or even surpass it. Secondly, early product introduction increases peak sales. If a product is made early it can gain and keep a large market share. Finally, a new product has a higher profit margin than a product that is produced by many competitors.

2. *More variants*: Versatile products require more parts for additional features and functions. Versatile products can also be customized to fit the personal needs of customers. A manufacturing system is forced to produce more product variants to meet the personalized needs of customers.

3. *Low and fluctuating volumes*: The required volumes of many products are reduced due to various reasons. Small market niches exist with many global competitors which reduces the volumes required from one company. The life cycle of new products are shorter and the durability of products have increased. Product customization has caused market demand to be fragmented into smaller portions.

4. *Low price*: There are many large companies that can produce products of the same quality at a reduced price. The price of a product is one of the primary features that customers consider before buying a product. Smaller companies are forced to find cheaper and more cost effective ways to produce products. The price of new products are also time dependent because once all the competitors have a similar product the customer only considers price when buying products.

DMSs and FMSs meet some but not all of the critical requirements for a manufacturing system to be successful in the modern market. Hence it is necessary to develop a manufacturing system that can meet all of the critical requirements. The proposed solution is a Reconfigurable Manufacturing System (RMS).

Authors agree that there is not a universal definition for RMSs. This is due to the fact that a system can be reconfigured on many levels inside the organization. For the purpose of this discussion and the rest of the research project, the following definition proposed by Bi *et al.* (2008) is used:

> *An RMS has an ability to reconfigure hardware and control resources at all of the functional and organizational levels, in order to quickly adjust production capacity and functionality in response to sudden changes in market or in regulatory requirements.*

In the definition *hardware* refers to machinery and *control resources* refer to software. For RMSs to work effectively, the machinery and software must be designed at the outset to be reconfigurable. This requires hardware and software modules that can be integrated quickly and reliably (Koren *et al.*, 1999). If this requirement is not met, the reconfiguration process will be lengthy and impractical.

Koren *et al.* (1999) proposes some key characteristics of RMSs to meet the requirements of the modern market:

1. *Modularity*: All major components must be modular. They include structural elements, axes, controls, software and tooling.

2. *Integrative*: The modules are designed with interfaces for component integration. Correct integration is required to make the whole system reconfigurable. Without it, only parts of the system can be reconfigured. The software interfaces are especially important.

3. *Customization*: This characteristic has two aspects: customized flexibility and customized control. Customized flexibility is when the machines used are designed around the part families that are produced. Extra flexibility, like in the case of CNC machines, is not required, thereby reducing costs.

Customized control has the same principle as flexibility but focusses on providing open ended software modules that only exert the required control. Extra functionality is not required, but the software is designed in such a way that extra functionality can be added on at a later stage, thereby saving money.

4. *Convertible*: The system must be able to change or convert its operating procedure on the fly. Conversion is required when the system changes from producing one type of part to the next within the same part family. Conversion requires the changing of tools, part-programs, fixtures, and even system capacity.

5. *Diagnosable*: Parts of poor quality need to be detected effectively in order to reduce the number of defects. Early detection prevents wasting further production resources on a defective part.

Using the definition of Bi *et al.* (2008) and the key characteristics proposed by Koren *et al.* (1999) the concept of RMSs can be explained. An RMS is created by using basic process modules, both hardware and software, that can be rearranged or replaced quickly and reliably (Mehrabi *et al.*, 2000). A system of this nature can provide customized flexibility for a particular *part family* and not only a part. The system is open-ended meaning that it can be improved, upgraded, and reconfigured, rather than be replaced (Mehrabi *et al.*, 2000).

For the purpose of this research, an RMS can be seen as an intermediate paradigm between a Dedicated Manufacturing System (DMS) and a Flexible Manufacturing System (FMS) (Bi *et al.*, 2008). The objective of the RMS being to provide the functionality and capacity that is needed, when it is needed (Mehrabi *et al.*, 2000). The machines used in the RMS as well as the control software are specifically designed for reconfigurability. The system must be able to quickly adjust capacity as well as functionality in response to demand changes.

Figure 2.1 is an illustration of the cost to construct a manufacturing system versus the capacity the manufacturing system can deliver. The DMS operates at a constant, planned maximum capacity. The only way to increase its capacity is to build another expensive manufacturing line. The capacity of the FMS is scalable at a constant rate because the capacity can be increased or decreased by adding or removing machines in parallel, but this is also an expensive solution. The capacity of a RMS is scalable at a non-constant rate depending on the initial design of the system and the market demand. The modular design of the system gives it the ability to adjust its capacity at a non-constant rate by adding or removing modules to meet market demand. Being able to adjust capacity to meet market demand will ensure that the system operates at full capacity most of the time, increase efficiency and reduce costs (Koren *et al.*, 1999).

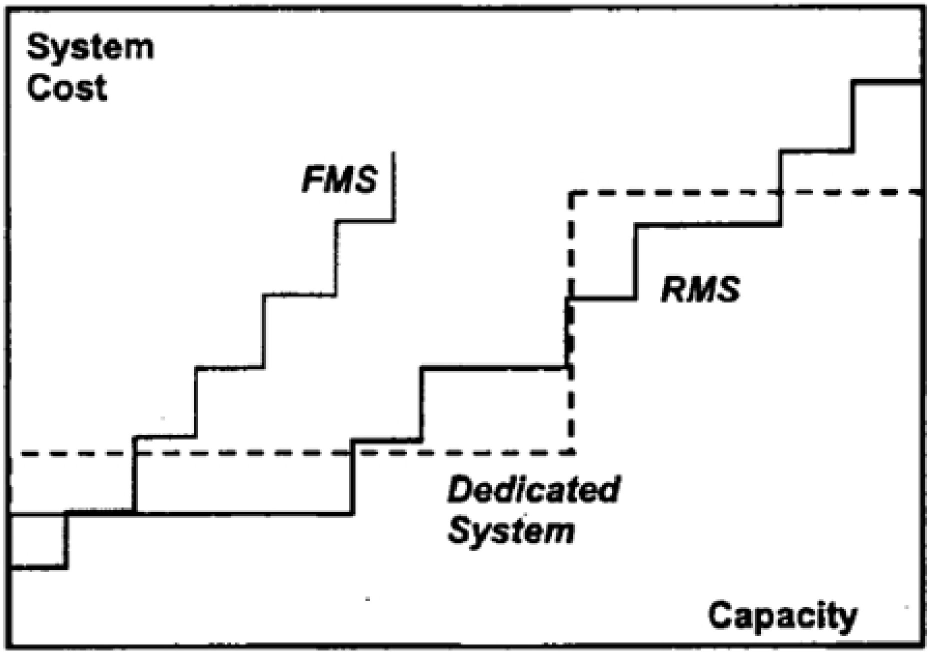


Figure 2.1: Manufacturing system cost versus capacity (or production rate) (Koren *et al.*, 1999).

In this section, the requirements of the modern market was discussed as well as the characteristics of RMSs that make these systems capable of performing in the modern market. Product variation was emphasized with shorter lead times and fluctuating volumes. RMSs are designed specifically for part families that can be used to assemble a wide variety of products.

A RMS is appealing to small and medium sized companies because the system can start off relatively small with a low capacity. As demand for the product being produced increases, capacity can be added quickly because of the modular nature of the machines and control software. Since the machinery have customized flexibility, the initial capital investment will be lower than purchasing flexible CNC machines. One drawback of a RMS however is that in house expertise in control systems and machine design may be required to implement and operate the RMS. This type of expertise may not be readily available and external help will be required, which could be expensive.

2.2 Part Families

According to Abdi & Labib (2004) manufacturing and the market were traditionally seen as two different environments to be studied. Time has proved that traditional and conventional manufacturing systems i.e. DMSs and FMSs cannot keep up with the dynamically changing market. Abdi & Labib (2004) proposes that in order to fulfill the gap between dynamic market demands and capacity and functionality of manufacturing systems, a reconfiguration link is necessary that group products into families, as shown in Figure 2.2.

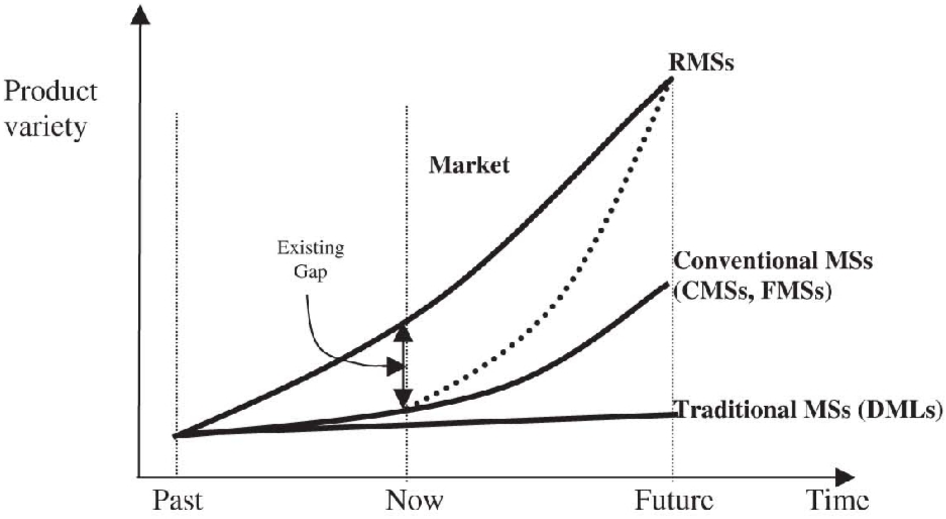


Figure 2.2: Effects of market changes on MSs over time (Abdi & Labib, 2004).

2.2.1 Part Grouping

In the reconfiguration link proposed by Abdi & Labib (2004), the product types that will be manufactured must first be selected based on market demand and available technology. The selected products for the product range can then be designed to have a modular structure i.e. products assembled from modular parts. Without modular parts, reconfiguration of the manufacturing system will not be possible. Hence it is important that the product and the process needs to be modular in order to achieve reconfigurability.

The design of modular parts and machines is beyond the scope of this project. It is however important to be able to group parts into families once they have been designed. According to Tam (1990) parts can be similar based on two characteristics:

1. Parts can have the same *machinery resource requirements.*
2. Parts can have *similar patterns of production sequences.*

The problem with grouping parts into a part family based on the first characteristic is that although the two parts require identical machines, the process of manufacturing the part can differ a great deal (Tam, 1990). An example of this is that two parts can both use a CNC machine but the machine uses different tools on each part respectively. Each tool change results in a ramp-up time that reduces the efficiency of the system.

The second characteristic appears to be a better solution to part grouping especially when keeping reconfigurability in mind. Tam (1990) proposes that each part and machine must be associated with a set of operations. The link between a part and a machine is then an indirect one through the operations performed on a part. There may be more than one machine that is capable of performing the required operation on the part. The choice of machine can now be based on how the parts are grouped.

2.2.2 RMSs and Part Families

The importance of RMSs becomes clear when producing part families. According to Xiaobo (2000) traditional and conventional manufacturing systems i.e. DMSs and FMSs have shown their limitations when producing part families because the configuration of such manufacturing systems are rigid and fixed. The configuration of a system includes the arrangement of machinery, the operating logic of the system, and the tooling of machines. Unlike DMSs and FMSs, RMSs are able to change their configurations easily.

RMSs are able to manufacture more than one member of a part family at a time. The machinery could use a different operating logic or a different tool to perform the required activities on the different parts. The key is that the same type of operation is performed on all members of the part family, since this is how the parts are grouped together (Xiaobo, 2000).

As previously stated, members of a part family can be manufactured simultaneously on on a RMS, but the members can also be assembled individually in batches. Once again the RMS can undergo configuration changes between batches to add capacity or functionality to the system. The decision maker has to decide on the optimal configurations for producing the different members of a part family. This is a typical selection problem which is addressed in this research project.

The design of a manufacturing system begins by selecting a product to produce. Once the parts that make up the product are grouped into part families and the manufacturing process has been identified, the next logical step is to focus on the machinery required to perform the required manufacturing activities. As expected, RMSs use machines that differ from the machines used in DMSs and FMSs, which is discussed in the next section.

2.3 Reconfigurable Machines (RMs)

Katz (2007) states that a typical RMS will use a combination of dedicated machines, conventional flexible machines and a new type of machine called the *reconfigurable machine* (RM) in its production line. RMs only form part of the RMS and is not the sole requirement for a manufacturing system to be reconfigurable. For the system to be reconfigurable it must be able to produce a part family and change the capacity of the system on the fly. RMs simplify the reconfiguration process because they are designed for the purpose of reconfiguration.

A RM can be explained by comparing it to machines used in DMSs and FMSs. Machines used in DMSs are designed around a specific part for mass production. The machine is designed to perform a single operation with high repeatability, reliability and high productivity. This results in a relatively simple machine at low cost (Katz, 2007).

Machines used in FMSs are designed to be able to perform most operations in a flexible manner. Machines of this nature are computer numerically controlled (CNC) and can produce a wide range of parts given different part programs. Repeatability, reliability and high productivity are also required from these machines. For flexible machines to meet these requirement they become expensive (Katz, 2007).

RMs are designed for customized flexibility which means that the machines have the flexibility to produce one or more *part families*. The machine can perform a pre-designed set of operations for a particular part family with high repeatability, reliability and high productivity. The limited, customized flexibility results in the reduction of investment cost and a fast response time when products change (Katz, 2007). The machines also have the added benefit of scalable capacity of which an example is shown in Figure 2.3. A common base can be used to add or remove machining modules that will increase or decrease production capacity for that machine. Scalable capacity gives the production system the ability to utilize the production resources as much as possible because the capacity is scaled according to the order size.

It is important to note that there is a difference between *products* and *part families*. Products are assembled from different parts that are produced on machines. Part families are formed by parts that require the same production operations and not the same machines. It is possible that two different products can have parts that belong to the same part family. For example the pistons of a V8 and V6 engine can be made on the same machine, even if the pistons differ in size and diameter, because the same operation is required. The result is that a

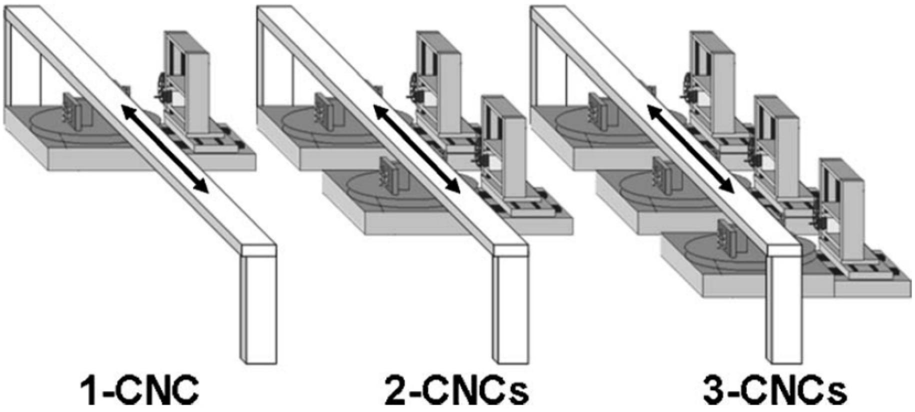


Figure 2.3: MultiCNC scalable Reconfigurable Machine (Spicer & Carlo, 2007).

V8 engine is a totaly different product from a V6 engine but one or more of their constituent parts can belong to the same part family.

Creating a machine for a specific part family will enable it to produce parts for many different products. The essence of a RMS is *having the flexibility to add new products to the production line and being able to produce it with the same machines. The only requirement is that the parts of the new product must be able to fit into the current part families of the production system.*

Katz (2007) has identified a number of principles for designing RMs. These principles are given to make it clear what is required to make a machine reconfigurable.

1. A RM is designed around a *part family*. This is the most important principle.
2. A RM is designed for *customized flexibility* only.
3. A RM must be able to be *converted rapidly*.
4. A RM must be *scalable*. This will allow the addition or removal of elements that can increase productivity or efficiency of operation.

5. A RM must allow reconfiguration so that it can work at several locations on the production line and be able to perform different tasks at each of these locations, using the same *basic structure*.
6. A RM should be designed to have a modular structure so that extra capacity or extra functionality can be added at a later stage.

The RMs are used to produce the required part families, but these machines on their own are not a system. A manufacturing system consists of many building blocks of which RMs and part families are only a part of. The system as a whole must be able to reconfigure which is discussed in the next section.

2.4 System Reconfiguration

In sections 2.2 and 2.3 part families and reconfigurable machines were discussed. Part families make up the primary building blocks of the products produced by the system. Grouping the parts of products into the correct part families is essential to ensure that the system can be reconfigured.

The machinery in a manufacturing line are the primary resources required to produce the parts of products. Without the correct arrangement of machines the system will not reach its goal of reconfigurability. People and control systems are secondary resources that utilize the primary resources to fulfill the needs of customers.

It has been established that parts are grouped into families based on the same manufacturing sequences. Manufacturing sequences define the flow of parts through the system, while the flow of parts refers to the movement of parts from one machine to the next in a sequential manner. Making the flow of parts through the system as effective as possible, will decrease the time of a manufacturing sequence and thereby increase the throughput rate of the system. Hence the optimal manufacturing sequence is dependent on the arrangement of the machinery.

A system can be reconfigured by changing the capacity of the system i.e. adding or removing machine modules. The system can also be reconfigured by adding functionality to the machines so that a new part family can be produced. If the layout of a system is fixed and the machinery cannot physically be moved, the system can be reconfigured using the concept of virtual system reconfiguration. The essence of virtual system reconfiguration is that by changing the manufacturing sequence of parts, different manufacturing lines can be created with the same machine layout. The details of virtual system reconfiguration are discussed in the following section.

2.4.1 Virtual System Reconfiguration

In order to explain virtual system reconfiguration, it is necessary to first develop a generic model of a system that will make virtual reconfiguration possible. Tang & Qiu (2004) have developed a generic model for a RMS and a representation of this model is shown in Figure 2.4.

From Figure 2.4 it is clear that the proposed RMS should consist of the following interacting and cooperating constituent units:

- Database.
- Inspection Unit.
- Supervision Unit.
- Shop Floor.
- Sensory Unit.
- Inventory.

According to Tang & Qiu (2004) the purpose and function of each of these units can be summarized as follows:

1. *Database*: The database is the central source of information for the system. The database can be subdivided into three parts consisting of the order database, process flow database and the resource database. The order database stores order information like quantity, product description, delivery date and other relevant information for the order. The process flow database stores the optimum manufacturing sequence of each part family for various manufacturing scenarios. The resource database keeps track

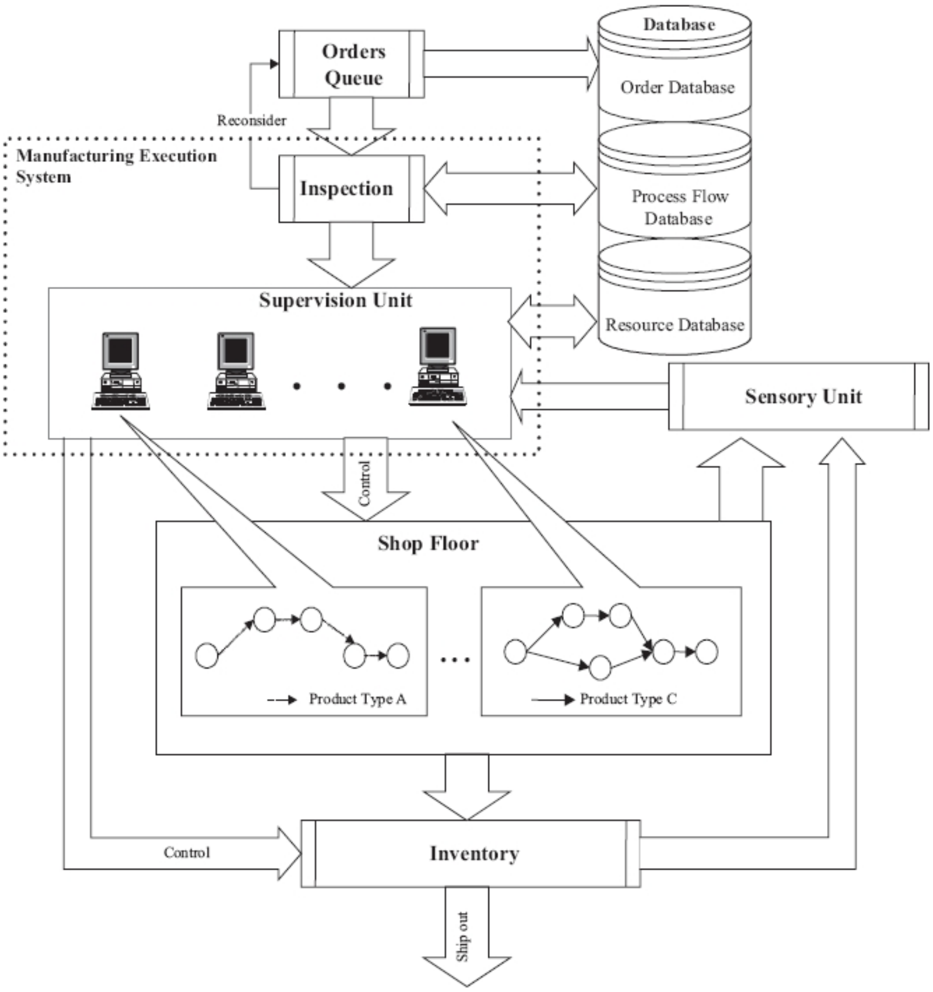


Figure 2.4: Generic Model for a RMS (Tang & Qiu, 2004).

of all the available machines, human operators, robots, inventory and the machine capabilities of the system.

2. *Inspection Unit*: The inspection unit will test different logical configurations of the system before the order is manufactured on the shop floor. The configurations are based on the order information and the shop floor status. From the order information, the inspection unit will determine which part families are required to fulfill the order. The shop floor status will include information like machine availability, Work In Progress (WIP) information, human operator availability and current production throughput

to mention a few. The optimum configuration of the system, for the selected part families, is chosen after which the optimum manufacturing sequence is determined from the process flow database to produce the various part families.

3. *Supervision Unit*: The supervision unit can be seen as the command center for the production line. The supervision unit will generally consist of software controllers, robot controllers and inventory controllers. There is also the so called super controller that communicates with all the other controllers in the system. An advanced information system is required to implement a supervision unit of this nature, but the enabling technology does exist. The main objective of each of the different controllers is to ensure that the planned manufacturing schedule and goals are met as effectively as possible.

4. *Shop Floor*: The shop floor is where all the facilities reside. This includes the machines, robots, materials, and human operators. In this proposed model of a RMS, the virtual configuration of the system to make a new order is formed logically and strategically on the shop floor according to the inspection unit result. Physical reconfiguration is done only once when all the equipment are installed. Orientating the machinery correctly is vital to make the virtual reconfiguration of the system economically viable.

5. *Sensory Unit*: The sensory unit is distributed all over the shop floor, consisting of various sensors and measuring equipment, to monitor the progress of production and give feedback to the supervision unit. The sensory unit plays an important role in the decision making of management by collecting real-time data from the shop floor.

6. *Inventory*: Inventory consists of all the finished products that are stored in a designated area until the order is completed. Once all the products for a single order are completed, they are packed and shipped to the customer. Inventory is controlled by the inventory controller in the supervision unit. Finding the correct balance of inventory is important because it represents "fixed" capital for the company. Work In Progress (WIP) and inventory are

costing the company money because of storage costs and it has not been sold yet. Having too much inventory usually leads to cash flow problems for companies so that policies like make-to-order, make-to-stock, JIT etc. are considered by the inventory controller to find the correct levels of inventory and WIP.

Figure 2.4 illustrates the proposed system units that are required to implement a system that can reconfigure by virtually changing its manufacturing lines. Virtual change is when the machines are not physically moved around on the shop floor, but rather the manufacturing sequence of parts are changed.

2.4.2 Virtual Manufacturing Lines (VMLs)

The concept of production routing has been used for many years. Currently Manufacturing Execution Systems (MESs) choose the optimal manufacturing sequence for each order. Once the manufacturing sequence is finalized, the MES will generate the relevant machine tasks and supervise production on the shop floor. As the materials move through the manufacturing sequences, shop floor personnel use the designated process flow, created by the MES, to monitor and control all the activities on the shop floor manually, semi-automatically or automatically. If there is a lack of accurate real-time data from the shop floor, the MES will display what is required by downstream operations and what machines are available based on the current information. It is then up to the production manager's discretion to select the correct machine from the available machines. This method results in many unnecessary setup times, slowed production and ultimately degraded system productivity (Tang & Qiu, 2004).

Product life cycles have shortened, global competition has intensified and the variation in demand has increased. These factors have had a great effect on the market environment. As a result it is not uncommon for manufacturers to receive orders that consist of a variety of products. The manufacturing environment has responded to the change in the market environment by increasing the amount of feedback and delivering real-time data from the shop floor. According to Mehrabi *et al.* (2000) in order for companies to deliver high quality products and services under these circumstances, companies need to adopt the make-to-order business model, which requires that their manufacturing lines be flexible, responsive and reconfigurable. Instead of having manufacturing lines predefined exclusively before production, lines need to be reconfigured concurrently based on the dynamics of order information and system status. It means that only once an order is placed and scheduled for production can a corresponding production line be logically, strategically and optimally formed (Tang & Qiu, 2004). The physical configuration of the machines on the shop floor do not change but the manufacturing lines on the other hand do change because they are virtual.

In a RMS, a Virtual Manufacturing Line (VML) is organized as a sequence of workstations that consist of one or more machines of the same type. It is then possible that the shop floor part flow diagram could be logically subdivided into multiple VMLs as shown in Figure 2.5.

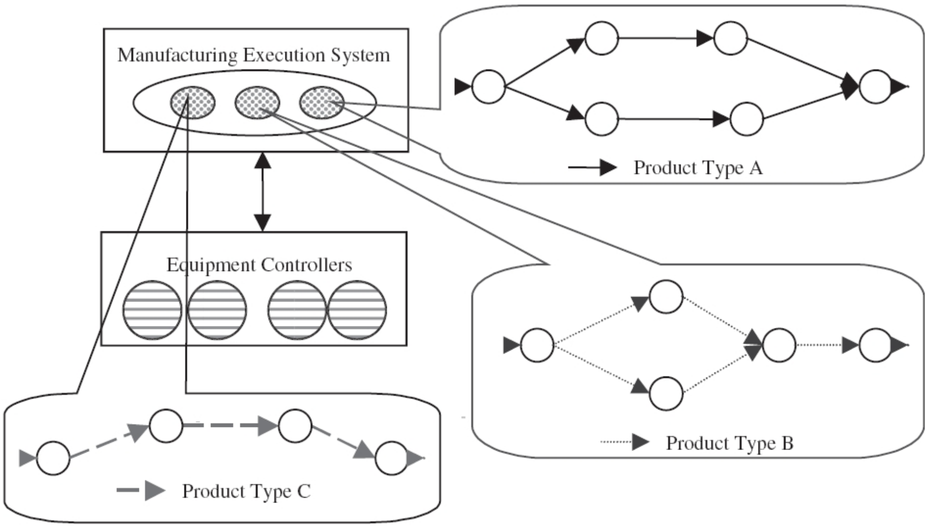


Figure 2.5: VML-based MES model (Tang & Qiu, 2004).

Each VML is dedicated to a part family and is controlled by a MES instance called a VML controller. The VML controller is created once the VML is formed logically. Once production of a part family has ended, the VML controller is revised for the next order or is dismissed. The advantage of this type of system is that it is modular, which means that if there is a problem with one VML, like a machine breakdown, only the relevant software and hardware modules are affected. The problem is isolated and can be modified without affecting the other VMLs. The result is that the manufacturing system as a whole becomes more flexible (Tang & Qiu, 2004).

Virtual system reconfiguration was discussed as an alternative to reconfiguring the system by physically adding or removing machine capacity and functionality. Whenever a system undergoes reconfiguration (virtual or physical), the system will lose some production time while the changes are made. The time lost due to reconfiguration forms part of the ramp-up time of the system, which is discussed in the following section.

2.5 Ramp-up Times

As stated in section 2.3 and shown in Figure 2.3 it is possible to create reconfigurable machines with scalable capacity. It is achieved by adding or removing modules from a common base. According to Spicer & Carlo (2007) the benefits of these type of machines include:

- Reduced capital investment.
- Reduced reconfiguration time.
- Reduced space consumption.

The set of system configurations that a scalable-RMS assumes as it changes over time is called its *configuration path*. For a manufacturer it is important to determine the configuration path with the least cost. Reconfiguration cost in this instance refers to when the capacity of the system changes by adding or removing equipment. If the reconfiguration cost of the system is not considered the problem becomes the standard capacity expansion problem. Standard capacity expansion problems determine how much and when to invest. Scalable-RMSs also look at how much and when to invest but it also determines in which system configuration to invest.

Reconfiguration cost includes (1) the cost of physical arrangement (labor cost), (2) the cost of lost capacity during system reconfiguration and (3) ramp-up costs.

An example of reconfiguration is shown in Figure 2.6 where two extra bases and an extra machine module are added in stage $k + 1$. This will greatly increase the flexibility of the system, due to the parallel operation of the system, and also expand its capacity. In stage $k + 2$ three bases and two machine modules are removed. Capacity of one type of machine can be reduced when new machines with better technology are installed. The older machines are still used to support the new machines. The reconfiguration cost when the system changes from producing one part family to the next will only take ramp-up costs into account.

According to Mehrabi *et al.* (2000) after the RMS is reconfigured, the production system must be "fine-tuned" before it can produce products consistently at the required quality standard and production capacity. The time required to do this forms part of the ramp-up time of the system. In general, the ramp-up time of the system is calculated as the sum of all the times that the system is idle while waiting for changes to be made to the system. Ramp-up time is calculated from the moment production stops until the time the system produces a completed part or product again. Mehrabi *et al.* (2000) state that the ramp-up time in traditional and conventional production systems i.e. DMSs and FMSs could take months or even years. For a RMS to be practical, ramp-up times need to be reduced significantly.

Figure 2.7 shows the relationships of the three phases of operation of a RMS when it is reconfigured to gain capacity. Most of the time the system operates in regular production mode at a constant capacity. When market demand changes, the system has to increase/decrease its capacity by adding/removing machines and equipment. This is called the reconfiguration phase, and there is a slight time delay before it can commence because equipment and machines need to be purchased. This procurement however does not affect regular production capacity. During reconfiguration, the system may lose capacity because manufacturing lines need to be shut down in order to install new machines. After the reconfiguration phase, the system goes through the ramp-up phase where it is "fine-tuned". The capacity of the system will gradually increase during this phase (Spicer & Carlo, 2007). Once all minor problems are solved, the system can operate at the new capacity. When a system is reducing its capacity there is also a reconfiguration phase but in this case the time delay is necessary to find a buyer for the old

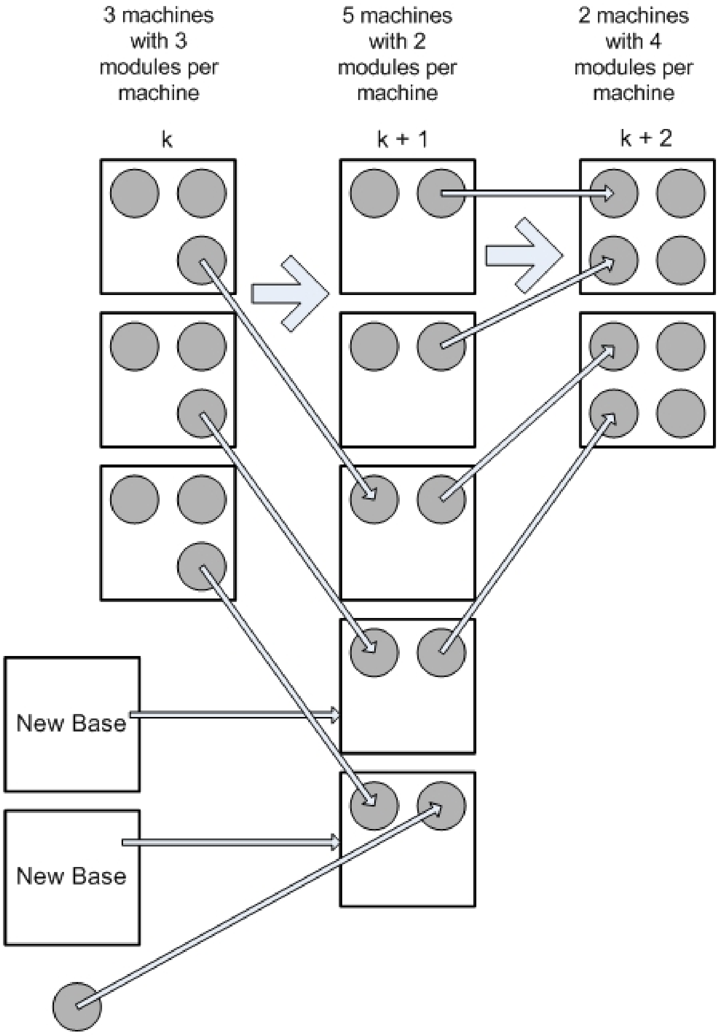


Figure 2.6: Process stage reconfiguration over several periods (adapted from (Spicer & Carlo, 2007)).

equipment. It will also not affect the regular production capacity and the rest of the the phases are the same except that the capacity is reduced instead of increased.

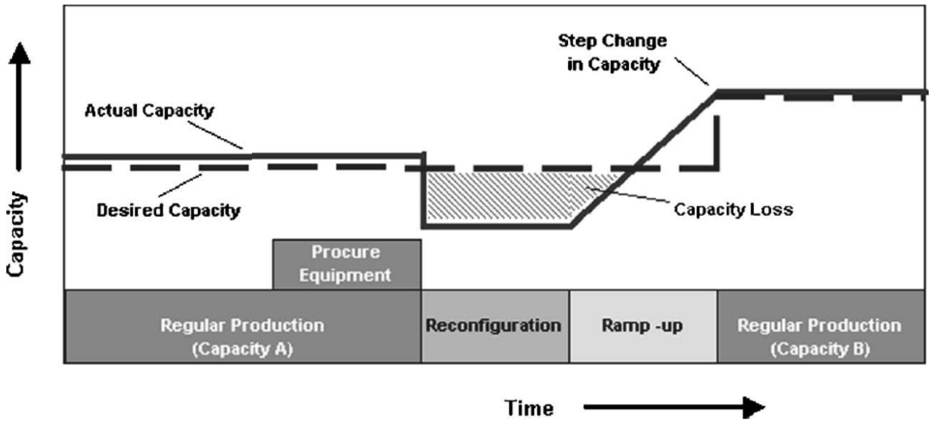


Figure 2.7: Phases of System Operation (Scaling up) (Spicer & Carlo, 2007).

2.6 Manufacturing and Simulation

Section 2.1 through to section 2.5 has given an overview of the world of manufacturing, and selected topics were discussed in more detail.

The concept of reconfigurability was explained and the need for it in the modern market was stressed. Conceptually a RMS sounds appealing, but it is important that the concepts must be tested before a reconfigurable system can be implemented. It is too expensive to build a real world system to test the concepts of reconfiguration. Therefore an analysis technique is required that is able to test the concepts of reconfigurability without the costs of actually building the system. Discrete event simulation was chosen as the analysis technique, since it is able to model a dynamic system at a relatively low cost.

2.7 Discrete Event Simulation

Discrete event simulation has increased in popularity as an analysis technique for dynamic systems due to the improvement of the available simulation software packages. Although modern simulation software is easy to use, there is more to simulation than just learning a program. In this section the principles of simulation will be explained as well as some modeling concepts. It is vital to understand these modeling concepts in order to build valid and credible simulation models. Simulation as an analysis technique is put into perspective with other analysis techniques and the advantages and disadvantages of simulation are also discussed.

2.7.1 Principles of Simulation

Simulation is a technique that has been used for many years to solve problems that are usually too complex to solve analytically. The principles discussed here are well known to simulation analysts but the terminology and concepts are explained because it will be used extensively in the remainder of this document. The principles and their explanations are adapted by the author from the *Handbook of Simulation* (Banks, 1998).

2.7.1.1 Definition of Simulation

Simulation is the imitation of the operation of a real-world process or system over time. In order to imitate the working of a real-world process, it is required to generate an artificial history, that can be studied to be able to draw inferences concerning the operational characteristics of the system. If the simulation model imitates the operation of the real-world system to a predetermined degree of accuracy, it can be used to describe and analyze the behavior of the system, ask what-if questions about different scenarios and aid in the design of real world systems. Existing as well as conceptual systems can be modeled with simulation.

2.7.2 Modeling Concepts

There are several concepts underlying simulation that include system and model, system state variables, entities and attributes, resources, list processing, activities, and delays. These underlying concepts are used by computer software to create the model of the real-world system and to simulate the operation thereof. Understanding these concepts are required to build an accurate model and to use the software effectively. The definition of a discrete-event simulation model follows from these concepts.

2.7.2.1 System, Model and Events

A model can be defined as *the representation of an entity/object in a form other than the entity/object itself* or *it is the representation of an actual system* (Banks, 1998). If a model is only a representation of an actual system, it must have defined boundaries and conditions that it will adhere to. The boundaries and conditions must be formulated by the analyst in such a way that they are not too complex, but complex enough to answer the questions raised by the problem.

An event can be considered as an occurrence that change the state of the system. Events can be external or internal to the system and they are respectively called exogenous and endogenous events. Consider for example a simple service station where customers arrive, receive service and then exit the system. The arrival of a customer is an exogenous event because it happens outside of the system being simulated. The arrival does however have an impact on the system and must be taken into consideration. The start of service on a new customer is an endogenous event because the event is within the boundaries of the system being simulated.

A discrete-event simulation model differs from other modeling techniques like mathematical models, descriptive models, statistical models and input-output models because it attempts to represent the internal components of a system and their interactions to such an extent that the objectives of the study are met. Other models represent the inputs and outputs of a system explicitly but represent the internal components only as mathematical and statistical relationships. These relationships are only based on theory whereas a discrete-event simulation model will include a detailed representation of the actual internals.

A discrete-event model is dynamic which means that the passage of time is considered in the model. Most mathematical and statistical models are static which means that they only represent the system accurately for a fixed point in time. A manufacturing system does change over time and it is one of the reasons discrete-event simulation was chosen as an analysis technique instead of using a mathematical or statistical model.

2.7.2.2 System State Variables

The system state variables are the collection of all the information required to define the events that occur within the system. System state variables are functions of the desired outputs or required results of the simulation study. The result is that two sets of state variables can differ for the same physical system depending on the purpose of the investigation. Determining the correct system state variables is as much an art as it is a science but with experience it becomes clear if there are any omissions or unnecessary variables.

System state variables are defined differently for discrete and continuous models. In a discrete-event model the system state variables will remain constant over intervals of time and only change in value at well-defined points called *event times*. In a continuous model the variables need to change continuously over time and not only at discrete points. In order to achieve this, the variables are defined by differential of difference equations that may change continuously over time.

2.7.2.3 Entities and Attributes

An entity is a representation of an object that requires explicit definition and about which information is gathered. Entities can be classified into two groups, as follows:

1. *Static*: A static entity does not move through the system and it is used to serve other entities. An example of a static entity is a cashier in a shop that serves customers.
2. *Dynamic*: A dynamic entity moves through the system by arriving at the system, receiving a service and exiting the system again. An example of a dynamic entity is the customer that the cashier serves.

All entities have attributes that describe the characteristics of the entity and are used to govern the behavior of the entity in the simulation model. Attributes should be considered as local values because an attribute can be of interest in one investigation and not the other. For example customers can be male or female and this attribute can be used to determine which gender buys the most of a single product. If the investigation only wants to determine the total sales of a product, the gender of the customer will not be of importance. From the example it is clear that many entities can have the same attribute or attributes i.e. many customers can be *male*.

2.7.2.4 Resources

As dynamic entities move through the system, they will usually receive a service from an entity called a resource. The resource can serve more than one dynamic entity at the same time when it is operating as a parallel server. Dynamic entities can request more than one unit of a resource if required.

If the request is denied, the dynamic entity will join a queue to await service or take some other action like divert to another resource or exit the system. Queues are also called files, chains, buffers and waiting lines. A resource can be in many different states but the most basic states are *busy* and *idle*. Other states also exist that include *failed*, *blocked* and *starved* to mention a few. The resource can only be in one state at any point in time and the possible states are defined by the analyst during the simulation study.

2.7.2.5 List Processing

As stated in section 2.7.2.4 entities are assigned to resources in order to receive a service. Assigning entities is done by attaching them to event notices and thereby suspending their activities into the future. The entities can also be placed on ordered lists which represent the queues in the simulation model.

The lists are processed by the resources according to predefined rules like FIFO (first in, first out), LIFO (last in, first out), the value of an attribute, or randomly to mention a few. An example of using the value of an attribute is when the list is processed according to SPT (shortest processing time) of the entities. The processing time is stored as an attribute of the entities and is then used to determine the order of the queue.

2.7.2.6 Activities and Delays

An activity is an action that takes place in the simulation model. The importance of the activity in simulation is not the action but the time it takes to complete. Hence an activity is defined as *a period of time whose duration is known prior to the commencement of the activity.* The advantage of knowing the time required to complete an activity gives the analyst the ability to schedule activities. The duration of activities can be determined as follows:

- *Constant*: The service of each entity can be a constant, e.g. five minutes, every day. It is a rare occurance because most operations have a time variance.
- *Statistical distribution*: The inter-arrival time of customers can be a random value from an exponential distribution with a mean of ten minutes.
- *Equation*: The service time could be 0.6 times a constant value from clock time zero to clock time five hours or 1.2 times a constant value after clock time 5 hours.
- *Input from file*: A situation can exist where the service time is five minutes when the preceding queue contains at most three entities and four minutes when there are four or more entities in the preceding queue.

A delay is also an action that takes place in the simulation model but contrary to an activity, the duration of a delay is unknown. Delays are caused by a combination of system conditions. An example is when an entity joins a queue, the waiting time is initially unknown since the waiting time depends on other events that may occur in the system. When an entity joins a queue that is processed according to LIFO, the waiting time is initially unknown because new entities can still enter the system.

Discrete-event simulation contain activities that cause time to advance and also delays that cause entities to wait. The important principle is that the beginning and ending of an activity or delay are events.

2.7.2.7 Discrete-Event Simulation Model

Modeling concepts have been introduced in the previous sections and it is now possible to define a discrete-event simulation model. *A Discrete-Event Simulation Model is a model in which the state variables change only at those discrete points in time at which events occur.* Events occur as a consequence of activity times and delays. Entities will compete for the available system resources and will possibly join a queue in order to do so. Activity and delay times will "hold" entities for durations of time and prevent them from moving forward in the system until the next event.

A discrete-event simulation model is executed over time by a mechanism that moves simulated time forward. Simulated time is a representation of real-world time that can be speeded up or slowed down by the analyst on the computer. The state of the system is updated at the end of each event, along with capturing and freeing of resources that may occur at that time.

2.7.3 Simulation in Perspective

Figure 2.8 shows where simulation fits into the modeling picture, based on the nature of the system being studied. There are three aspects of simulation that need to be considered when studying a system. The first is the time dependency of the system. If the system is independent of time, it is called a *static* system and when time does play a role it is called a *dynamic* system.

The second aspect is the characteristics of the variables which can be *deterministic* or *stochastic*. A deterministic variable's value is known or can be calculated whereas a stochastic variable's value is of a random nature.

The third aspect is the simulation time increment that governs how the values of variables change during the simulation study. As stated in section 2.7.2.2 the system state variables can be discrete or continuous. The value of discrete variables stay the same for periods of time and only change value at each event time. The value of continuous variables is a function of time because they are defined by differential or difference equations.

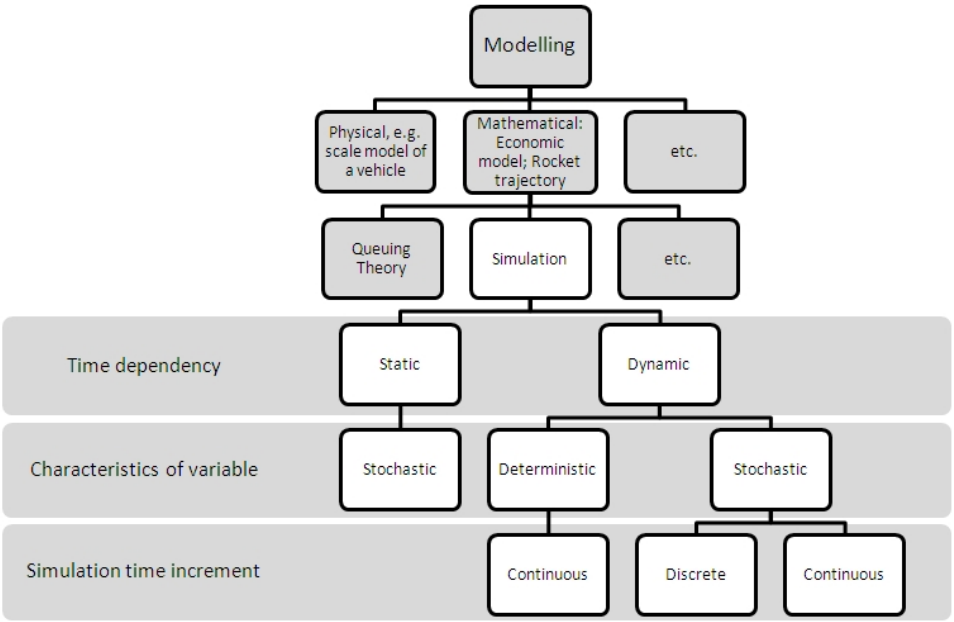


Figure 2.8: Simulation in perspective (Bekker, 2009).

2.7.3.1 Why simulate?

The first step in any study of a system is to identify the problem. Once the problem has been identified the analyst can start to investigate analysis techniques that will not only solve the problem satisfactorily but do it in a cost effective way. There are many analysis techniques available like queuing theory, linear programming, assignment algorithms, integer programming, and dynamic programming to mention a few (Bekker, 2009). Some of these techniques will try to solve the problem analytically with the aid of a computer. When the problem is of such a complex nature that it cannot be solved analytically, simulation will be strongly considered by the analyst. In addition, if the problem is of a complex stochastic nature, then simulation is again indicated as a suitable analysis technique.

The arrivals of orders in a manufacturing system are of a stochastic nature and hence it can be represented as a random variable in a statistical distribution. In South Africa there is still a large human contribution in the manufacturing environment and fully automated manufacturing lines are a rare occurrence. Humans are prone to commit errors and this will also increase the variation in the manufacturing system.

A manufacturing system is dynamic because all the activities are dependent on time. The state of the system will change as time advances but these changes happen at discrete points. Analytical methods will only represent the system at a fixed point in time while a simulation model can take the passage of time into account. The discrete event times can be determined by the analyst and makes it possible to build schedules into the simulation model.

It has been established that when a system is of a complex stochastic nature, has variation in the process, and is dependent on time, simulation is strongly considered as a analysis technique. If simulation is indeed chosen to analyze the system, it is important to understand the following when doing the study and when interpreting the results:

- The system being studied operates in the real world which is a continuous environment. The computers that are used to run the simulation model, that represent this continuous environment, do calculations in a discrete environment. The result is that the values obtained from the computer can never be a hundred percent correct. It can only be stated with a certain level of confidence that the true value of a result is covered by a certain numerical range. This range of values is called a confidence interval which will be discussed in detail in a subsequent section.

- In order to create a model, which is a representation of a real-world system, assumptions need to be made. Assumptions are made to simplify the problem so that it can be formulated and programmed into the computer. Assumptions are clearly stated during the study and must be validated before the model is constructed. Assumptions cause the model to lose some fidelity but because they are validated, the model does not lose any credibility.

2.7.4 Advantages and Disadvantages of Simulation

Competition in the computer industry has lead to technological breakthroughs concerning computer hardware. The result is that companies can now produce computers cost effectively with more computing power and larger memories. The improvement of hardware has a direct relation to the improvement in available software.

As simulation software improve, more and more business are starting to realize the benefits of using simulation. Businesses are also starting to use simulation in their daily activities and not only as a once off design aid. Banks (1998) has compiled a list of the advantages and disadvantages of simulation, that many authors have identified, and they are the following:

2.7.4.1 Advantages

1. *Choose correctly*: In a situation where a proposed change or addition to a system is considered, simulation can be used to evaluate different alternatives without the commitment of resources to the problem. In other words, no hardware has to be purchased or systems have to be installed without testing it thoroughly with simulation. This is critical because once hardware is purchased or a system is installed, it is difficult and expensive to change it.

2. *Compress and expand time*: Simulation gives the analyst the ability to speed up or slow down phenomena in order to investigate them thoroughly. An entire shift can be examined in a matter of minutes while hours can be spent on analyzing all the activities that occurred during a single minute of simulated activity.

3. *Understand why*: Managers often want to know why certain phenomena occur in a real system. Simulation aids in the answering of "why" questions by reconstructing the scene and taking a microscopic examination of the system to determine why the phenomena occurs. Simulation is used because it is difficult to see and control the real system in its entirety.

4. *Explore possibilities*: Once a valid simulation model have been developed, it is possible to explore new policies, operating procedures, or methods without the expense and disruption of experimenting with the real system. This is one of the greatest advantages of using simulation software because when

modifications are incorporated into the model, the effects of the changes can be observed on the computer rather than on the real system. These modification can also be realized in a matter of hours at minimal cost while it could take weeks to implement them in the real system at a great expense.

5. *Diagnose problems*: The modern manufacturing and service environment have become so complex that it is nearly impossible for a single person to consider all the interactions taking place at a given moment. An analyst will spend many hours building a simulation model that will try to incorporate all the complexities of the system. The model will then help management to better understand the interactions among the variables that make up the complex system. Diagnosing problems are simplified because management have gained insight and have increased their understanding of the variables and their effects on the system's performance.

2.7.4.2 Disadvantages

1. *Model building requires special training*: Simulation software have simplified the actual programming and model building but it is still strongly advised to get a well trained simulation analyst to perform the analysis. It is not difficult to build a model but to build a *valid* model that correctly represents the real world system takes training and experience. Furthermore, if two competent simulation analysts each build a model of the same system, there will be similarities but it is highly unlikely that the models will be exactly the same. This does not however reduce the credibility of the individual models because different assumptions can be made in both cases.

2. *Simulation results may be difficult to interpret*: The inputs of simulation models are usually random variables from statistical distributions which result in random output variables. A sound statistical background is required to be able to distinguish whether outputs are caused by randomness or system interrelationships.

3. *Simulation modeling and analysis can be time consuming and expensive*: A simulation study will take time if it is properly executed. The study can

also become expensive in some cases but skimping on resources for modeling and analysis may result in a model that is inadequate to solve the desired problem.

4. *Simulation may be used inappropriately*: It takes a competent simulation analyst to realize that in some cases simulation is not the correct analysis technique for the problem and that an analytical analysis can give better results. An example is that some waiting lines can be analyzed by simulation but that queueing theory is an inexpensive or even better solution, at least for long-run evaluation.

The principles of simulation and the modeling concepts discussed in this section are the basic knowledge required by an analyst to be able to build a simulation model. Building the model is only part of the analysis because once the model is built, it will generate output data. A true simulation analyst must be able to statistically analyze the output in order to reach meaningful recommendations. In the following section, the statistical analysis techniques that were used in this research project, are explained.

2.8 Statistical Analysis of Discrete Event Simulation Models

Modern simulation software has simplified the process of modeling a real world system accurately. Irrespective of the software that is used to build the model, a sound statistical analysis of the output data is required in order to reach meaningful conclusions.

According to Law (2007), a common mistake made in simulation studies is that only a single simulation run of arbitrary length is made and the results of this run is then treated as the ”true” model characteristics. The problem with this approach is that simulation models use random variables sampled from probability distributions to drive the model through time. The output from one simulation run is just a particular realization of random variables that may have large variances. As a result, the estimates from the output of one simulation run could differ greatly from the corresponding true characteristics of the model.

Hence there is a significant probability of making erroneous decisions based on the output of a single simulation run.

2.8.1 The Random Nature of Simulation Output

Since we use stochastic variables to drive a simulation model, it is fair to say that the output gathered from a simulation run is of a random nature. Let $Y_1, Y_2, \ldots$ be an output stochastic process from a single simulation run. For example, Y_i can be the number of customers present in a shop during the ith hour. All the Y_i's are random variables that are in general not independent of each other nor are they identically distributed.

A simulation run of length m is made (in the example it will be m hours), using the random numbers $u_{11}, u_{12} \ldots,$ where the ith random number used in the jth run is denoted as u_{ji}. The result from the simulation run is that $y_{11}, y_{12}, \ldots, y_{1m}$ are a realization of the random variables $Y_1, Y_2, \ldots, Y_m$. If a different set of random numbers $u_{21}, u_{22} \ldots,$ is used for another replication, then a different realization $y_{21}, y_{22}, \ldots, y_{2m}$ is obtained for the random variables $Y_1, Y_2, \ldots, Y_m$.

Using this logic, a number of n replications (runs) of the same length m are made. For each replication, different random numbers are used from the same probability distribution to drive the model. Each replication has the same initial conditions and the statistical counters used to calculate the performance measures are reset at the start of a replication. In general, the following observations are the result:

$$\begin{array}{ccccc}
y_{11}, & \ldots, & y_{1i}, & \ldots, & y_{1m} \\
y_{21}, & \ldots, & y_{2i}, & \ldots, & y_{2m} \\
\vdots & & \vdots & & \vdots \\
y_{n1} & \ldots, & y_{ni} & \ldots, & y_{nm}
\end{array}$$

The observations from a particular replication (row) are not Independent and Identically Distributed (IDD). However, the observations from the ith column $y_{1i}, y_{2i}, \ldots, y_{ni}$ are IDD observations of the random variable Y_i, for $i = 1, 2, \ldots, m$. Thus there are *independence across replications* which is the key to doing output data analysis (Law, 2007).

2.8.2 Types of Simulation Output Analysis

According to Altiok & Melamed (2007), simulations can be either terminating or nonterminating, depending on whether there is an obvious way for determining the run length. A terminating system is when an event E exists that specifies the run length or replication length. From the previous section it is clear that the random variables are IDD across replications and can be compared. For terminating systems, the event E can be specified in a number of ways. If a system starts in an empty state, the event E that terminates the run can be when the system is once again empty. The event E can also be at a point in time beyond which no more useful information can be gathered or the event E can simply be specified by management or the simulation analyst.

A simple example is a shop that only does business during office hours. At the beginning of a day, there are no customers in the shop and the system is empty. During the day, customers will arrive according to a probability distribution and will exit the system once they have received service. At the end of the day, the shop closes and the system is once again empty when the last customer leaves. Hence there is an event i.e. ”the shop closing” that will end the simulation run. A business day like this will be replicated a number of times using different random variables that change the arrival and service times of customers. Some of the performance measures can be the service level or the average number of products sold per hour.

For a nonterminating simulation, there is no natural or predefined event E that can specify the run length of a single replication. Nonterminating simulations are used when the natural operating characteristics ”over the long run” of a system need to be determined. Any system of this nature has a transient phase followed by a steady state phase. Techniques have been developed to determine when the system has reached steady state and the appropriate run length required to be able to make meaningful recommendations. An example of a nonterminating system is a communications network. One problem with nonterminating systems, is that the system changes over time. The communications network can increase in size or more users can be added to the network that will affect the capacity. A simulation study of this nature will have to be modified or repeated continuously as the system changes over time.

In this research project, a number of different manufacturing systems will be investigated with the help of simulation. These manufacturing systems have the characteristics of a terminating simulation model because the system starts at an empty state, does some operations and ends in a empty state again. There is also an event that causes the simulation run to terminate when all the orders are completed.

It can be argued that a manufacturing system is a nonterminating system because these type of systems usually run for unlimited time periods and only stop for maintenance or breakdowns. This is true, however it was decided to model the systems under study for this research project as terminating systems. The systems can be modeled in this way because the simulation experiments were specifically set up to be terminating.

2.8.3 Analysis of Terminating Systems

In any simulation model the value of a certain *parameter*, call it θ, will be desired. The simulation will be programmed to produce an *estimator*, $\hat{\Theta}$, for the true but unknown parameter, θ, which evaluates to some *estimate*, $\hat{\Theta} = \hat{\theta}$ (Altiok & Melamed, 2007). The distinct meanings of the entities θ, $\hat{\Theta}$, and $\hat{\theta}$ are as follows:

- θ is a deterministic but unknown parameter and it could possibly be a vector.
- $\hat{\Theta}$ is a *variate*(random variable) estimator of θ.
- $\hat{\theta}$ is a realization of $\hat{\Theta}$.

For each replication r of the simulation model, the estimator $\hat{\Theta}$ yields a different estimate, $\hat{\Theta}(r) = \hat{\theta}(r)$. For example, suppose that θ is the (unknown) average waiting time of customers waiting in a queue to receive service. The estimator $\hat{\Theta}$ of θ can then be the sample mean of the waiting times, $\{X_1(r), \ldots, X_n(r)\}$, where $X_j(r)$ is the jth customer waiting time observed during replication r and n is the number of customers that waited (Altiok & Melamed, 2007).

In simple terms, estimators of true parameters are used because it is impossible to do an infinite number of replications to find the true value of a parameter. However it is possible to do "enough" replications to be able to estimate the true value of a parameter with a certain level of confidence.

2.8.3.1 Estimating Means and Variances

In section 2.8.1 it was explained that variables are IID across replications. This result is useful, because when these variables are sample means, the Central Limit Theorem can be utilized to conclude that these variables (Y_i) are approximately normal distributed. If a sample of n replications are used (assuming n is enough), a sample mean can be calculated which is an unbiased *point estimator* of the *parameter* μ (population mean). The sample mean can be calculated using equation 2.1 and the sample variance which is also an unbiased *point estimator* for the population variance can be calculated using equation 2.2 (Law, 2007).

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n} \tag{2.1}$$

$$S_{\bar{X}}^2 = \frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n-1} \tag{2.2}$$

2.8.3.2 The Confidence Interval

The result of equation 2.1 is an unbiased *point estimator* of the population *parameter* μ. Since it is only a point estimator, the need arises for a more descriptive or informative estimator. This has given rise to the calculation of an *interval estimator* called the confidence interval. The confidence interval specifies a range in which the unknown population parameter is to be expected. Suppose the parameter to be estimated is θ, then the interval estimate of θ is $[L, U]$ so that $P(L \leq \theta \leq U) = 1 - \alpha$. In this case $(1 - \alpha)$ is called the confidence level, and α is the level of significance which is the probability that the confidence interval will not include the population parameter. For example, if 100 confidence intervals are constructed, with a confidence level of 95%, then we expect that 95 out of the 100 intervals will contain the true population parameter. This is referred to as the coverage of the confidence interval (Altiok & Melamed, 2007).

Figure 2.9 illustrates the coverage of a confidence interval. It is clear that the estimated sample distribution includes the true population parameter, but there is still an error between the sample mean, $\bar{x}$ and the true mean, μ.

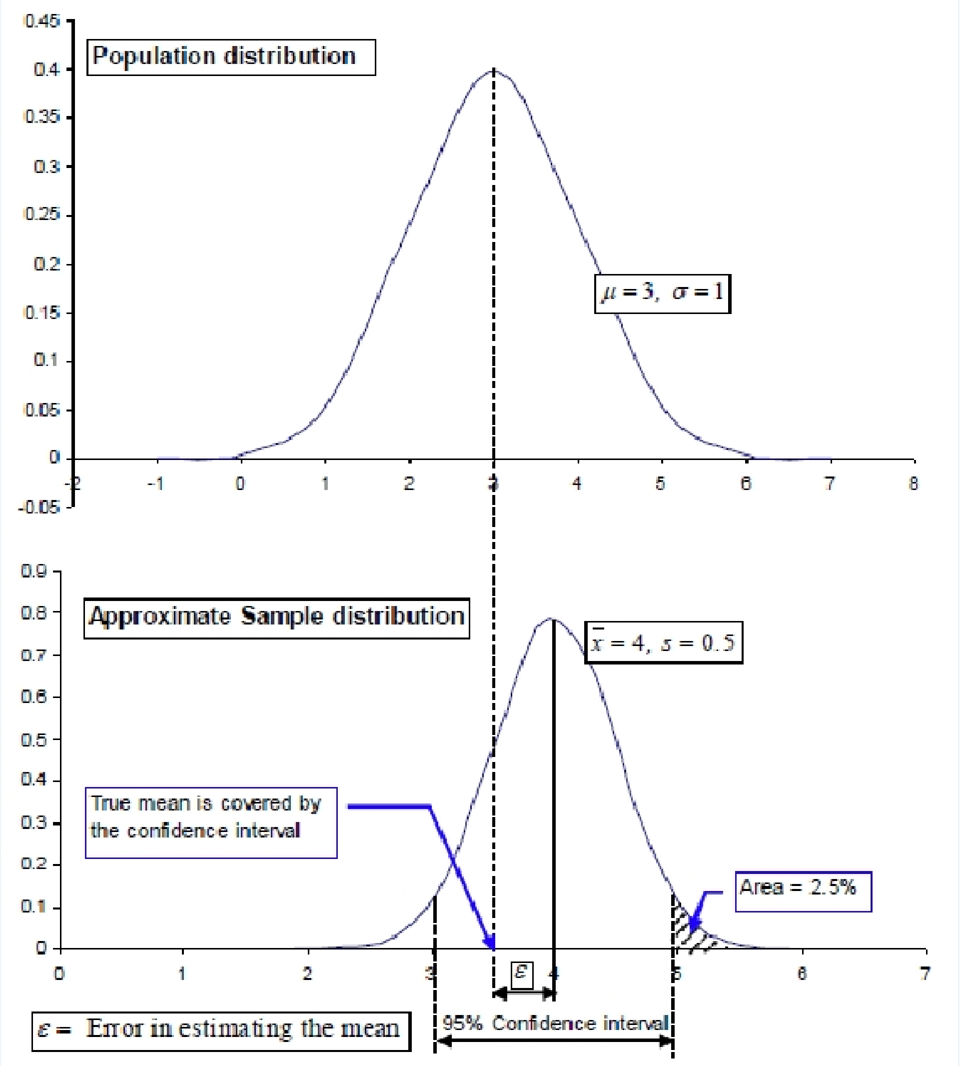


Figure 2.9: Confidence Interval Coverage (Bekker, 2009).

As stated in the previous section, the samples from across the replications are approximately normal distributed due the the Central Limit Theorem. The confidence interval $P(L \leq \theta \leq U) = 1 - \alpha$ can then be written as:

$$P(-z_{1-\frac{\alpha}{2}} \leq \frac{\bar{\Theta} - \mu}{\sqrt{\frac{\sigma^2_{\bar{\Theta}}}{n}}} \leq z_{1-\frac{\alpha}{2}}) = 1 - \alpha \tag{2.3}$$

where $z_{1-\frac{\alpha}{2}}$ is the upper critical value of the cumulative normal distribution.

In equation 2.3, $\bar{\Theta}$ is estimated using equation 2.1. Also in equation 2.3 the population variance σ is used, which is not known when doing simulation studies. Hence the population variance must be estimated with the sample variance using equation 2.2. Since the sample variance is calculated, the Student t-distribution has to be used to calculate the confidence interval. Instead of using the random variable Z, the random variable T is calculated with equation 2.4.

$$T = \frac{\bar{\Theta} - \mu}{\sqrt{\frac{S^2}{n}}} \tag{2.4}$$

The random variable T has a Student t-distribution with $n - 1$ degrees of freedom. The new confidence interval can be constructed using equation 2.5.

$$P(-t_{n-1,1-\frac{\alpha}{2}} \leq \frac{\bar{\Theta} - \mu}{\sqrt{\frac{S^2}{n}}} \leq t_{n-1,1-\frac{\alpha}{2}}) = 1 - \alpha \tag{2.5}$$

Equation 2.5 can also be written in the form of equation 2.6:

$$\begin{aligned} CI &= \bar{\Theta} \pm h \\ &= \bar{\Theta} \pm t_{n-1,1-\frac{\alpha}{2}} \sqrt{\frac{S^2}{n}}, \qquad n \geq 2 \end{aligned} \tag{2.6}$$

where h is referred to as the half-width and $t_{n-1,1-\frac{\alpha}{2}}$ is the upper $1 - \frac{\alpha}{2}$ critical value from the Student t-distribution with $n-1$ degrees of freedom. Note that for a fixed α, the half-width is a measure of accuracy for the confidence interval estimator. The smaller the variance, the narrower the half-with will be. Since $z_{1-\frac{\alpha}{2}}$ and the variance σ^2 are fixed, it is clear from equation 2.3 that the width of the confidence interval can only be reduced by increasing the number of replications, n. The same reasoning is valid when using the Student t-distribution for constructing the confidence interval (Altiok & Melamed, 2007). Intuitively it makes sense because if more replications are made, the closer the sample comes to representing the entire population.

2.8.4 Comparing Systems Statistically

The confidence intervals that were constructed in the previous section indicate how much the estimator differs from its true value. When comparing systems with each other, it is required to construct a confidence interval for the *difference* between the two systems. For this application a paired-t confidence interval is constructed to determine whether two systems differ based on a certain output parameter.

It is good practice to verify one statistical testing method with another one to ensure that the results obtained are indeed correct. A variation in a box plot can also be used to test whether there is a statistical significant difference in the output of systems. For example, there are two layout designs for a new manufacturing plant and it is required to examine the difference based on throughput rate.

2.8.4.1 A Paired-t Confidence Interval

The paired-t confidence interval is used to compare two systems with each other. It can however be modified to compare more than two systems which will be discussed in a subsequent section. For the paired-t confidence interval, a number of n_1 replications must be made for system 1 and n_2 for system 2, where $n_1 = n_2$. The output parameters X_{1j} and X_{2j} can be paired to form a new variable $Z_j = X_{1j} - X_{2j}$, for $j = 1, 2, \ldots, n$. Note that the X_{ij}'s must be normal distributed. The Z_j's are IID random variables and $E(Z_j) = \zeta$, the quantity for which the confidence interval is constructed (Law, 2007). Equations 2.7 and 2.8 can be used to calculate the mean and variance respectively.

$$\bar{Z}(n) = \frac{\sum_{j=1}^{n} Z_j}{n} \tag{2.7}$$

$$\widehat{Var}[\bar{Z}(n)] = \frac{\sum_{j=1}^{n}[Z_j - \bar{Z}(n)]^2}{n(n-1)} \tag{2.8}$$

From equations 2.7 and 2.8, the approximate $100(1-\alpha)$ percent confidence interval can be calculated:

$$CI = \bar{Z}(n) \pm t_{n-1,1-\frac{\alpha}{2}} \sqrt{\widehat{Var}[\bar{Z}(n)]} \tag{2.9}$$

If the Z_j's are normal distributed, the confidence interval will cover ζ with probability $(1-\alpha)$. If not, the Central Limit Theorem can once again be applied to ensure that the confidence interval will approximately cover ζ for large values of n (Law, 2007). If the *difference* confidence interval includes zero, the two systems do not statistically differ. If the *difference* confidence interval includes only positive values, the output of system 1 is greater than the output of system 2. If the *difference* confidence interval includes only negative values, the output of system 2 is greater than the output of system 1.

2.8.4.2 Comparing more than two systems with a t-test

One method that could be used to compare more than two systems is when all the new designs are compared to a standard or an existing system. In essence only two systems are compared at a time because all the systems are still compared only to the standard and not each other. It makes sense to do it when an existing system must be modified or replaced. However, if there is no standard to compare to, all the designs must be compared to each other. This is called an all pairwise comparison.

The confidence-interval approach can still be used by making several confidence-interval statements simultaneously. According to Law (2007), the individual confidence levels will have to be adjusted upward so that the *overall* confidence level of all intervals' covering their respective targets are still at the desired level of $(1-\alpha)$. The Bonferroni inequality is used to ensure that the overall confidence is at least $(1-\alpha)$. The Bonferroni inequality implies that if a number of c confidence statements are made, each separate interval must be constructed at a level of $(1-\frac{\alpha}{c})$ to reach the overall confidence level.

If there are k different designs, confidence intervals will be constructed for the differences $\mu_{i_2} - \mu_{i_1}$, for all i_1 and i_2 between 1 and k, with $i_1 \leq i_2$. For this case there will be $\frac{k(k-1)}{2}$ individual intervals which means that each interval must be constructed at a confidence level of $1 - \frac{\alpha}{[\frac{k(k-1)}{2}]}$.

2.8.4.3 Comparing more than two systems with SMORE plots

The confidence intervals calculated in the previous sections focus on the spread of values around the mean. This can give meaningful results but the total spread of values of an output variable must also be considered when comparing systems to each other. Simio (the simulation software used in this project) includes a new type of chart for reporting on output statistics, called the Simio MORE or SMORE plot (W. David Kelton, 2010). SMORE plots are a combination of an enhanced *box plot*, first described by John Tukey (John W. Tukey, 1978). SMORE plots are based on the *Measure Of Risk and Error* (MORE) plots developed by Barry Nelson (Nelson, 2008). Figure 2.10 illustrates the components which make up a SMORE plot.

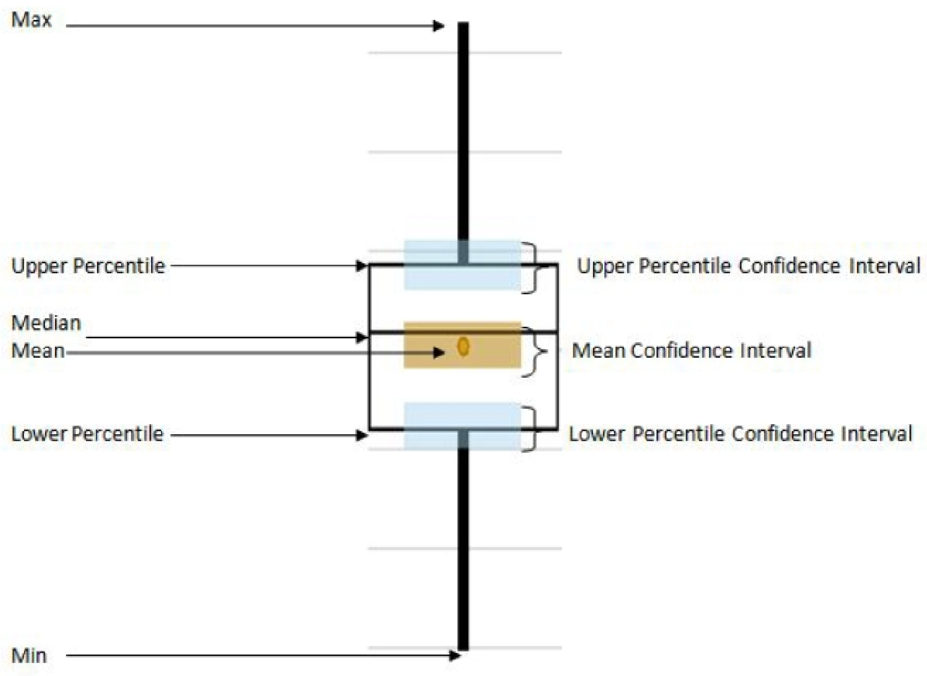


Figure 2.10: SMORE plot components (W. David Kelton, 2010)

It is similar to the box plot of Tukey because it displays the maximum and minimum observed values, the sample mean, median, and the lower and upper percentile values. The percentile values can be specified but the default value is at 25% for the lower and 75% for the upper percentile (W. David Kelton, 2010). Confidence intervals are also calculated for the upper and lower percentiles to evaluate the risks associated with the simulation output (Nelson, 2008).

In order to compare systems to each other a variation on the box plot of Tukey, called the *notched box plot*, is used (John W. Tukey, 1978). A *notch* which is a confidence interval around the *median*, and not the *mean*, must be calculated. The standard deviation of the median, s can be calculated using Equation 2.10 where R is the interquartile range and N is the number of observations or simulation replications completed.

$$s = \frac{1.25R}{1.35\sqrt{N}} \tag{2.10}$$

The notch around the median may be calculated as

$$M \pm Cs \tag{2.11}$$

where C is a constant. If the standard deviations of the systems that are being compared are similar, a value of $C = 1.386$ can be used (John W. Tukey, 1978). Hence the notches can be calculated using Equation 2.12.

$$M \pm 1.386(\frac{1.25R}{1.35\sqrt{N}}) \tag{2.12}$$

The result is a range for the specified output variable. Two or more systems can be compared to each other using these calculated notches. It will be explained with the use of an example. Suppose system 1 has notch of (6,9), system 2 a notch of (4,5), and system 3 has a notch of (7,10). In this case system 1 and system 2 differ statistically because their notches do not overlap. It is also clear that system 1 has a higher output for the variable under consideration. However, system 1 and system 3 do not statistically differ because their notches overlap. System 3 and system 2 do differ with system 3 having the higher output. Hence if the notches of systems overlap, they do not differ and vice versa when the notches do not overlap (John W. Tukey, 1978).

2.9 Multi-objective Optimization

Finding a solution to a problem is not always sufficient. In most problems the goal will be to optimize the outcome by finding the best solution. Optimization refers to finding the best possible solution for a problem according to the constraints of the problem. For single objective problems, the best possible solution is called the global optimum. For complex problems it might not be possible to find the exact global optimum, but a good approximation can give meaningful results (Coello Coello, 2006).

When a problem has more than one objective that needs to be optimized simultaneously, the problem is called a multi-objective problem (MOP) (Teleb & Azadivar, 1994). According to Coello Coello (2006) the objectives of a MOP are normally in conflict with each other. If this is not the case, a single solution exist for the MOP because the objectives can then be optimized in a sequential order until the global optimum is found. A simple example of conflicting objectives in manufacturing are maximizing the quality of a product but at the same time minimizing the production cost of the product. There is not a single answer that can be regarded as the global optimum for this type of problem. MOPs have a set or a vector of solutions, where each solution is a trade off between the objectives. This set of solutions is called the *Pareto optimum* and the term was coined in 1896 by Vilfredo Pareto. The Pareto optimum is defined as *the solution to a MOP if there exists no other feasible solution which would decrease some criterion without causing a simultaneous increase in at least one other criterion* (Coello Coello, 2006). The solutions that are in the Pareto optimal set, are non-dominated, because there are no other solutions that are better for the particular values of the constraints and input variables.

The formal mathematical definitions for the Pareto optimal set and the general form for a MOP are given to clarify the meanings of each. According to Bekker & Aldrich (2011), the MOP, in general, is a problem of the type:

$$
\begin{aligned}
&\textit{Minimize} && \\
&f(\mathbf{x}) := [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})] && (2.13)\\
&g_i(\mathbf{x}) \leq 0, i = 1, 2, \ldots, p && (2.14)\\
&h_i(\mathbf{x}) = 0, i = 1, 2, \ldots, q && (2.15)
\end{aligned}
$$

where the vector of decision variables is denoted by $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$, $f_i : \mathrm{I\!R}^n \to \mathrm{I\!R}$, $i = 1, 2, \ldots, m$, are the objective functions, while the constraint functions are $g_i, h_j : \mathrm{I\!R}^n \to \mathrm{I\!R}, i = 1, 2, \ldots, p; j = 1, 2, \ldots, q$.

Since MOP usually have at least two conflicting objectives, there exist many acceptable solutions for a given problem. These form the *Pareto optimal set.* A few definitions pertaining to Pareto optimality are necessary and they are:

Definition 1 Given two vectors $\mathbf{u}$ and $\mathbf{v} \in \mathrm{I\!R}^m$, we say that $\mathbf{u} \leq \mathbf{v}$ if $u_i \leq v_i$ for $i = 1, 2, \ldots, m$, and that $\mathbf{u} < \mathbf{v}$ if $\mathbf{u} \leq \mathbf{v}$ and $\mathbf{u} \neq \mathbf{v}$.

Definition 2 Given two vectors $\mathbf{u}$ and $\mathbf{v} \in \mathrm{I\!R}^m$, we say that $\mathbf{u}$ dominates $\mathbf{v}$ (denoted by $\mathbf{u} \prec \mathbf{v}$) iff $\mathbf{u} < \mathbf{v}$.

Definition 3 A vector of decision variables $\mathbf{x}^* \in \mathcal{F}$ ($\mathcal{F}$ is the feasible region) is *Pareto optimal* if there does not exist another $\mathbf{x} \in \mathcal{F}$ such that $\mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}^*)$.

Definition 4 The *Pareto optimal set* $\mathcal{P}^*$ is defined by $\mathcal{P}^* = \{\mathbf{x} \in \mathcal{F} | \mathbf{x}$ is Pareto optimal$\}$.

Definition 5 The *Pareto front* $\mathcal{P}^*_T$ is defined by $\mathcal{P}^*_T = \{\mathbf{f}(\mathbf{x}) \in \mathrm{I\!R}^n | \mathbf{x} \in \mathcal{P}^*\}$.

Solving an MOP requires that the Pareto optimal set be found from the set of all decision variable vectors that satisfy Equations (2.14) and (2.15).

The equations and definitions also hold if the goal of the problem is to maximize the solution instead of minimizing it.

The logic followed when doing multi-objective optimization can be explained with the help of Figure 2.11. The range for decision variables are determined according to the constraints of the problem. The values of the decision variables are varied inside the specified range to obtain different realizations of the objective functions. For example, one combination of x_1 and x_2 results in a value for both objectives, f_1 and f_2. Hence many variations in the decision space will result in

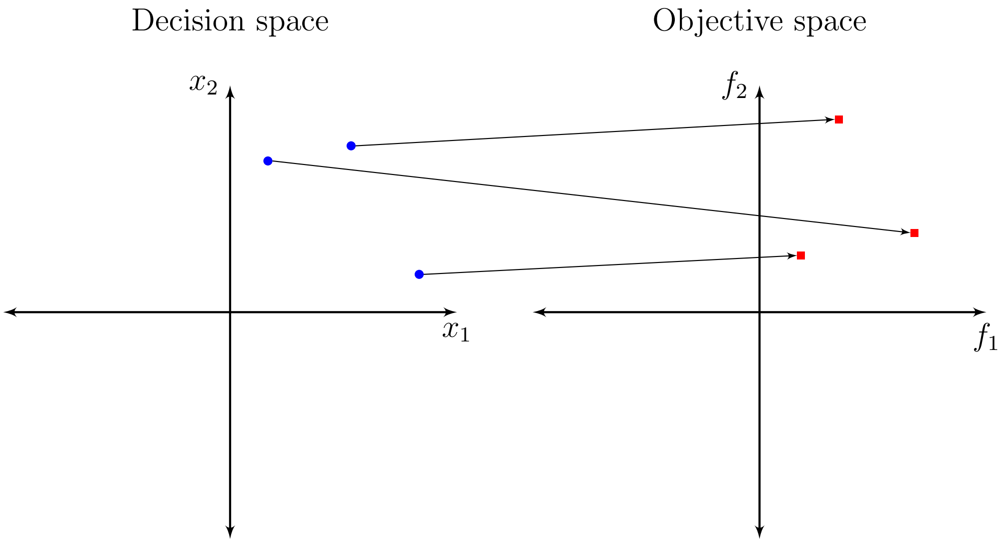


Figure 2.11: MOP Mapping.

many possible combinations of f_1 and f_2 in the objective space. The objective space can then be used to calculate the Pareto front.

In this research project, different manufacturing systems will be studied and tested under various conditions. Due to the complex nature of manufacturing systems, there are many conflicting objectives that could be optimized simultaneously. Hence, it is possible to to have many combinations of variables in the decision space that result in realizations in the objective space. Therefore, a manufacturing system is a perfect application for the use of multi-objective optimization.

With the knowledge gained in this chapter, it is possible to formulate the research problem for this research project.

2.10 Research Problem Description

This chapter has given background information about manufacturing systems and how Reconfigurable Manufacturing Systems (RMS) came to be what it is today. The use of RMS for the production of part families was emphasized as well the system's ability to change capacity quickly. Since RMS are fairly new to the manufacturing industry, studies are required to better understand the effects of reconfiguration on the system.

It was established that manufacturing systems are too complex to analyze analytically with a mathematical model. Hence simulation was chosen as the analysis technique and a thorough background of simulation as well as some statistical analysis techniques were discussed. With an understanding of the field of study as well as a chosen analysis technique, the problem that will be investigated can be explained as follows:

The Department of Mechanical and Mechatronic Engineering and the Department of Industrial Engineering at Stellenbosch University, as well as researchers from the Central University of Technology (CUT) in Bloemfontein are working together on a research project. The project is for one of the initiatives of the Department of Science and Technology, called AMTS (Advanced Manufacturing Technology Strategy). The AMTS has a need for research in the field of Reconfigurable Manufacturing Systems and has appointed the three already mentioned institutions to form a partnership and work together on the project.

The project can be divided into three distinct research fields, which are machine design, control system design, and simulation analysis. The Department of Mechanical and Mechatronic Engineering is responsible for the machine designs, CUT for the control system design, and the Department of Industrial Engineering for the simulation analysis.

A simple conveyor assembly system will be built to test the designs. The conveyor system will be required to assemble five different parts from a single part family and be able to change its capacity quickly. The resources used to assemble the parts are pallet magazines, part feeders, welders, and inspection & removal stations.

The aim of the simulation analysis is to aid the designers of the machines and the control system to better understand the working of the system. The conveyor assembly is expensive to build, which is why simulation is so valuable during the design process. Different configurations can be evaluated on the computer before actually building the final design. However, in this project a conveyor system was donated to the Department of Mechanical and Mechatronic Engineering which means that the configuration of the conveyor is fixed due to financial constraints. Since the conveyor configuration is fixed, the system will not be able to change capacity because there is physically no space to add extra machinery. All the machines necessary to assemble the parts, for example the welding station, will not be procured for the study since a machine like it costs in excess of a million rand. These constraints do not affect the simulation study.

The following steps will be executed in this study to contribute to the overall project:

- Build a model of the current conveyor system with all the required machinery to study the operating characteristics of the system.
- Build alternative configurations with different conveyor layouts and routing procedures to search for an optimal configuration.
- Allow the alternative configurations to change capacity, and assess the effects. Capacity changes has an effect on the machine design, but it has a major effect on the control system.
- Optimize each configuration and give the designers an idea of what is possible and to answer as many what if questions as possible.

It is important to emphasize that simulation studies are used to inform decision makers, not prescribe to them what to do. Although simulation models are not a hundred percent correct, they should give a good indication of what outputs can be expected given certain inputs. The decision still lies with the decision makers.

The aims of this study are as follows:

- To develop a better understanding of the characteristics of the RMSs that will be developed during this study.
- To apply the concepts of multi-objective optimization to a manufacturing problem and thereby optimize the RMSs that were developed.

- To create a decision support system, in the form of simulation models, for the machine and control system designers, that could be used to test their designs.
- To develop a platform for future researchers on which different reconfigurable systems can be developed and other system characteristics can be tested.

In the next chapter, the methodology that was followed to develop the simulation models are discussed. Each alternative configuration is also explained in more detail as well as the verification and validation that was performed on the proposed designs.

CHAPTER *3*

MODEL DEVELOPMENT

In the previous chapter some background information about manufacturing systems and simulation was given. The knowledge gained made it possible to develop simulation models to test the concepts and characteristics of reconfigurable manufacturing systems. In this chapter the methodology that was followed to develop these simulation models will be discussed.

This chapter will focus on the selection of simulation software and how it was used to model the existing conveyor configuration of the Department of Mechanical and Mechatronic Engineering. From this initial conveyor configuration, three alternative conceptual designs were developed and modeled with simulation.

The operating logic of the configurations are explained together with a detailed description of how each alternative design was developed. The chapter concludes with a discussion of how the models were validated and verified.

3.1 Simulation Software

In a simulation study, it is important to understand the system that needs to be modeled in order to build a credible model. Once the logic of the system is documented and the assumptions for the model are justified, it is time to build the model in a simulation software package. If the model is designed correctly, it should be possible to build the model in any simulation software.

Simio was chosen because it is a modern, state-of-the-art object-orientated simulation package, and a free, fully licensed package was donated to the Department of Industrial Engineering by Simio LLC, for education and research purposes.

3.2 The Parts

Before a manufacturing system can be designed, the part that will be manufactured must first be known. One aim of this research project is to evaluate a RMS and study the characteristics thereof. Hence not only a suitable part is required but rather a suitable part family. The reason for this is that RMSs are perfectly equipped to handle part families as was explained in section 2.1.3 in chapter 2.

The part family was selected by the Department of Mechanical an Mechatronic Engineering at Stellenbosch University. They have been working for years on developing a reconfigurable assembly system that will automatically assemble the Q-frame breaker mechanism of a Q-frame circuit breaker. These products are currently being assembled manually by a manufacturing company in Johannesburg. The Q-frame breaker mechanism can be seen in Figure 3.1 and an internal view of an assembled Q-frame circuit breaker can be seen in Figure 3.2.

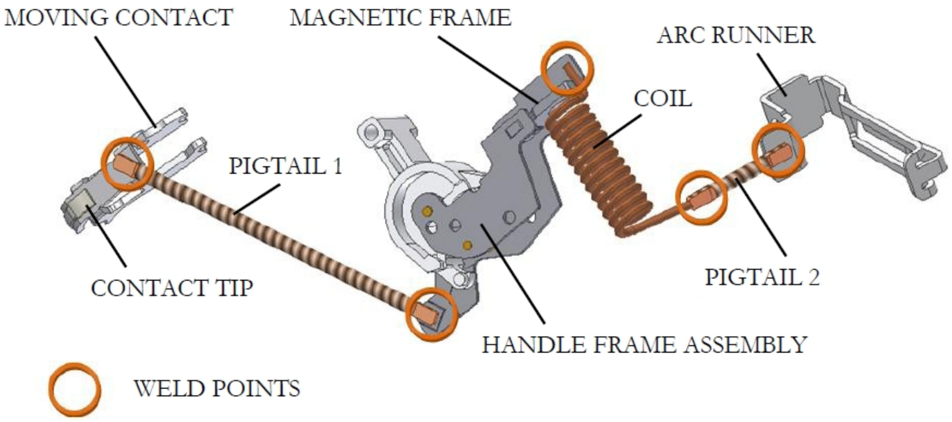


Figure 3.1: Q-frame breaker mechanism assembly (Sequeira, 2008).

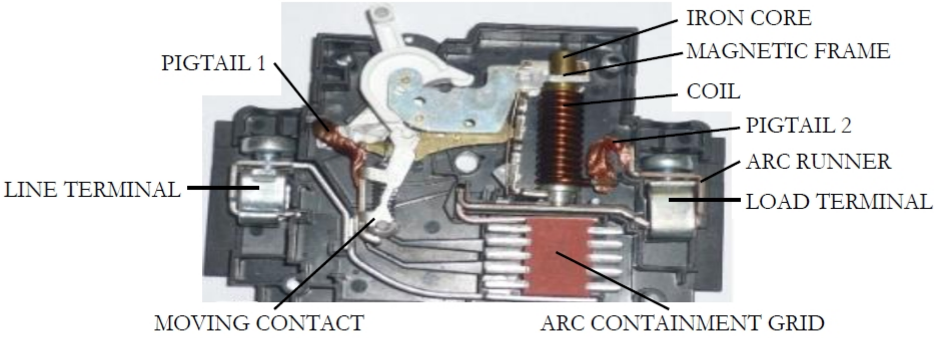


Figure 3.2: Internal view of assembled Q-frame circuit breaker in closed switch position (Sequeira, 2008).

There are eleven part variations for the Q-frame breaker mechanism and each variant correspond to a different Ampere rating. The coils and two pigtails (twisted copper braids with nuggetized ends) vary geometrically between the models, while the moving contact, arc runner and handle frame assembly only vary in plating and material (Sequeira, 2008). For this research project only five variants of the Q-frame breaker mechanism were used and this was deemed sufficient to test the reconfigurability of the system.

3.3 System Modules

As stated in section 2.1.3 in chapter 2, one of the key characteristics of a RMS is modularity. All the major components of the system should be modular, which include structural elements, axes, controls, software, and machinery. The modules that are used to construct a RMS also need to be customizable so that the system has the flexibility to manufacture or assemble a part family.

The layout of the initial conveyor configuration is shown in Figure 3.3 to give an indication of how the system modules are used to assemble the Q-frame breaker mechanisms.

The basic modules that are used to assemble the Q-frame breaker mechanism part family are:

- Conveyor system

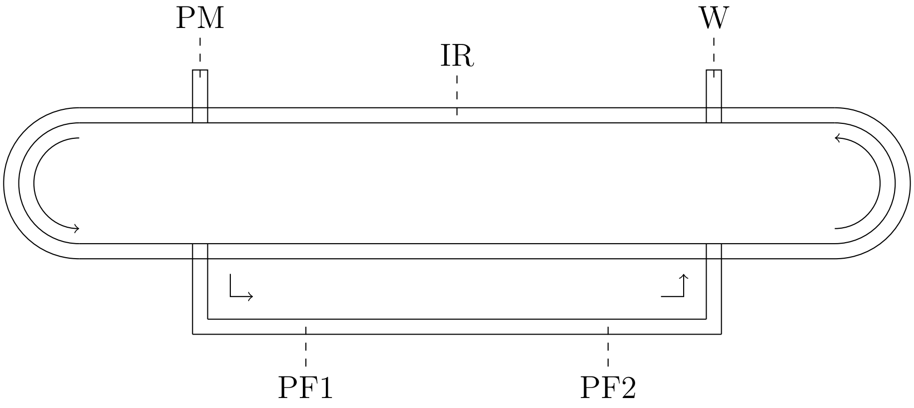


Figure 3.3: Initial conveyor configuration.

- Pallets
- Pallet Magazine
- Part Feeder
- Welder
- Inspection & Removal

The abbreviations used in Figure 3.3 are as follows:

- PM - Pallet Magazine
- PF1 - Part Feeder Type I
- PF2 - Part Feeder Type II
- IR - Inspection & Removal
- W - Welder

From Figure 3.3 it is clear that the pallets circulate in a counter clockwise direction. The system starts in an empty state after which pallets are placed onto the system by the pallet magazine. The pallets receive parts from the part feeders, inspection is done at the inspection & removal station, and the required welds are completed at the welding station. These system modules will now be discussed in more detail.

3.3.1 Conveyor System

The conveyor system transports pallets to each of the work stations and enables the system to complete the necessary process steps to assemble the Q-frame breaker mechanism. Most conveyor systems are modular because they can be assembled to fit the route that is required to transport work. The design of the conveyor layout is an important step because once it has been installed, it is usually fixed. This does not mean that the layout cannot be changed, since it is comprised of modular pieces, but changing the layout is disruptive and time consuming. The Simio representations of all the system modules are given so that the reader can identify them when studying the simulation models. Note that these machines and modules will look different in reality but the focus of this research is on the simulation analysis. Hence the Simio representation of a conveyor module can be seen in Figure 3.4. In the figure a transverse conveyor module is also visible.

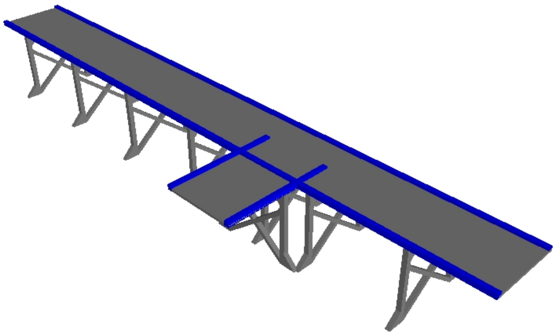


Figure 3.4: Conveyor Module: Simio representation.

The conveyor system is customizable because the speed of the conveyor can be changed to fit the needs of the assembly system. Once again this must be decided before construction because an increase in conveyor speed requires larger motors to drive the conveyor. Simulation is a useful tool to determine what conveyor speeds are required for the desired production levels.

3.3.2 Pallets

The conveyor system is used as a transportation method but for the application of this project, it cannot do it alone. Pallets are frequently used to transport goods with conveyors, especially when machine operations need to be performed during the production cycle.

For the Q-frame breaker mechanism, parts need to be placed by part feeders after which they are welded together. Hence the parts cannot be placed loosely on the pallet but require a fixture to hold the parts in place. A CAD representation of a pallet with a fixture can be seen in Figure 3.5. The fixture will differ slightly for each variation in the Q-frame breaker mechanisms while the pallet remains the same. Note in Figure 3.5 that the pallet is fixed with a RFID (Radio-frequency Identification) tag. These tags are used as mobile data carriers for pallet routing purposes. The pallet dimensions are $330mm \times 330mm$ which results in a conveyor width of $330mm$.

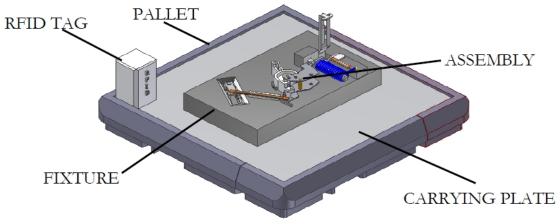


Figure 3.5: Pallet and fixture: CAD representation (Sequeira, 2008).

In Figure 3.6 the Simio representation of a pallet and fixture can be seen. Instead of explicitly showing the different fixtures, the pallets are given different colors to represent the different Q-frame breaker mechanisms.

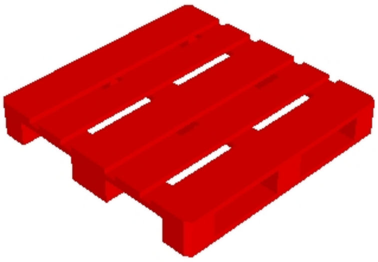


Figure 3.6: Pallet and fixture: Simio representation.

3.3.3 Pallet Magazine

The pallet magazine is a machine that places pallets on the conveyor system and can also remove the pallets from the conveyor system. The pallet magazine has a capacity of between 15 and 20 pallets. The pallet magazine consists of a steel cage mounted on top of a stand with a rotating pallet cartridge inside the cage. A transverse conveyor module is used to insert and remove pallets from the pallet magazine. The pallet cartridge has three stacks of pallets and can store pallets for all the types of Q-frame breaker mechanisms. This is possible because the pallet stays the same while only the fixture located on the pallet changes between the different frames. The Simio representation of the pallet magazine can be seen in Figure 3.7.

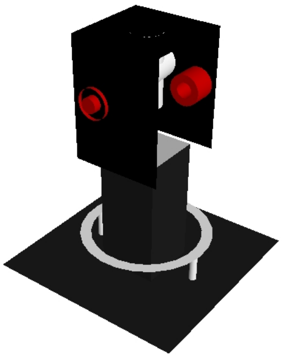


Figure 3.7: Pallet Magazine: Simio representation.

3.3.4 Part Feeder

At the time of writing, it was not yet established exactly how the part feeders will work in reality. Ideally the part feeder consists of feeding units that feed and align the parts, while a robot arm places the parts. The robot arm will be able to place more than one type of part. Hence the robot has customized flexibility which is a requirement for RMSs.

There are six parts that need to be placed on the fixture before the Q-frame breaker mechanism can be welded. As stated in section 3.2 the coils and the two pigtails vary geometrically across the range of Q-frame breaker mechanisms. If the shape of a part is different, it is almost like a totaly different part for a robot arm to pick and place. Hence with three parts changing in geometry for five part variations and two parts only changing in material, there are 17 different parts for the robot arm to pick and place.

For the simulation model it was deemed necessary to have two types of part feeders to distribute the work load. The one part feeder will pick and place a sub set of the 17 parts while the other part feeder will pick and place the remaining set of parts. This method is justified because it requires less customization per robot and it gives increased capacity to the system that will increase productivity. In the simulation models, part feeder type I is yellow while part feeder type II is red, which can be seen in Figure 3.8.

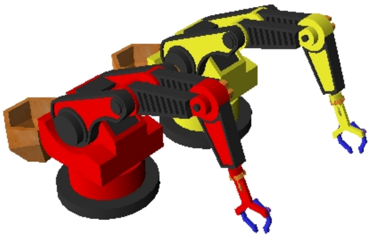


Figure 3.8: Part Feeders Type I and II: Simio representation.

3.3.5 Welder

The purpose of the welder is to fix all the parts together at the designated weld points as shown in Figure 3.1. The welds are completed using a spot welder with a moveable welding head. The spot welder is the most expensive system module and could not be procured for this study since it costs in excess of a million rand. However the Simio representation of the spot welder can be seen in Figure 3.9.

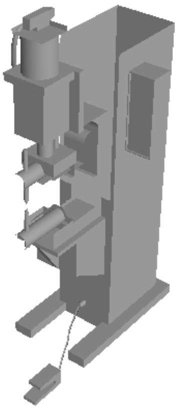


Figure 3.9: Spot Welder: Simio representation.

3.3.6 Inspection & Removal

As the name of this module states, there are two processes that happen when a pallet enters the inspection & removal station. The first process is to inspect whether the tasks performed by the other system modules were done correctly and the second process is to remove the parts from the pallet. The removed parts can be either good or defective. Once the parts are removed, the pallet will go back into circulation to start the assembly process again. The inspection process can happen at different times during the assembly process and will be discussed in a subsequent section.

The inspection & removal module consists of a robot arm with a camera fixed to it or mounted next to the robot. The camera is used in machine vision technology where the image of the parts on the pallets are compared to a benchmark image to determine whether it is defective or good. If the parts are assembled defectively, they are discarded or reworked while the correctly assembled parts will go into inventory. The Simio representation of the inspection & removal module can be seen in Figure 3.10.

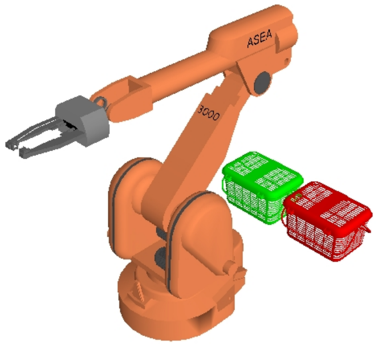


Figure 3.10: Inspection & Removal: Simio representation.

Note that the green bin represents correctly assembled while the red bin represents defectively assembled Q-frame breaker mechanisms.

3.4 Assembly Process

With the use of the system modules, discussed in section 3.3, the assembly process of the Q-frame breaker mechanism can be explained. Note that the Q-frame breaker mechanism will subsequently be referred to as the Q-frame.

The assembly process uses a conveyor system in conjunction with pallets to transport parts in the system. In each of the designs, the conveyor is essentially a loop that circulates the pallets. The result is that the conveyor has a maximum capacity for the number of pallets that can circulate at the same time. This makes it possible to determine the optimum number of pallets for a desired production rate.

The system starts in an empty state after which the pallet magazine places pallets onto the conveyor. Each of the pallets is equipped with a RFID tag which is used to route the pallets and identify which type of Q-frame will be assembled on the pallet. The RFID tag also communicates with the control system and relays information about which assembly steps the pallet has completed and which are still pending.

The basic assembly steps include part placing, inspection, and welding. Parts are placed on the pallet after which the placement is inspected in one of two ways. Once inspection is completed, the parts are welded together and inspected again. After the second inspection, the assembled part is removed and the pallet re-enters circulation. The details of the individual assembly steps are discussed in the following sections.

3.4.1 Part Placing

When the pallets are placed on the conveyor, they are empty and need to receive parts. Hence the logical first step is to visit the part feeders. The pallet needs to receive service from a part feeder type I and II in order to have all the required parts. It was not explicitly defined that the parts must be placed in a predefined sequence. *Sequence* means that the pallet has to receive service from a part feeder type I before receiving service from a part feeder type II. Another option is that the pallet just needs to visit both part feeder types regardless of sequence. Simulation models were built to test the effect of both cases.

3.4.2 Inspection Method

Once a pallet has received all the required parts, welds are applied to fix the parts together. A prerequisite for welding is that the placement of the parts must first be inspected before welding can occur. This inspection can be done in two ways, the first of which is done at the inspection & removal station. This requires the pallet to be routed to the inspection & removal station before the pallet can receive service from the welding station. At the inspection & removal station the placed parts are deemed good or defective and dealt with accordingly. Defectively placed parts are removed and put back into circulation for rework while good parts can continue to the welding station. If the parts were defectively placed, the pallet will once again be empty and need to return to the part feeders.

The second inspection method is when inspection is done at the part feeders while placing the parts. The result is that the pallets that leave the part feeders will all have correctly placed parts and are ready to receive service from a welding station. Another benefit is that a production step is skipped because the pallets no longer need to visit the inspection & removal station before welding. This inspection method is more expensive because more cameras are required for inspection purposes.

Once the Q-frame has been welded, it needs to be inspected again and removed from the conveyor so that the pallet can re-enter circulation. This final inspection is only done at the inspection & removal station because the Q-frame must also be removed from the pallet. The good Q-frames are used to assemble the final circuit breaker while the defective Q-frames are either scrapped or reworked. The reworking process is beyond the scope of this project.

3.4.3 Batch vs Mix

As stated in section 3.2 five Q-frame variants are assembled on a single system. The Q-frame variants are numbered from Q1 to Q5 and a standard order will demand all five. There is a large variation in the quantities demanded for each with a relatively high demand for frames Q1, Q2, and Q3 and a relatively low demand for frames Q4 and Q5.

With five part variants and a high variation in demand, production can be approached in two ways. The first is to do mixed assemblies where all five Q-frames are assembled at the same time. Since a conveyor configuration has a maximum capacity, the number of pallets assigned to each Q-frame depend on the proportion of their demand to the total demand. The advantage of this method is that individual orders will be completed in full before the next order is started. However the variation in order quantities for each Q-frame has an effect on the overall throughput rate of the system. The reason being that Q-frames with relatively small orders are assigned less pallets than the Q-frames with relatively large orders. With few pallets, the individual throughput rate of the particular Q-frame is low, which has an effect on the overall production rate of the system.

A second approach is to assemble the Q-frames in batches where only one variant is assembled at a time. Once a batch of one type of Q-frame is completed, the pallets are removed from the conveyor by the pallet magazine and pallets with other fixtures for the next batch are placed onto the conveyor. This has a delay implication but the result is that individual throughput rates of variants increase. Orders will most likely be pooled together to reduce the delay of batch changes. With good scheduling, this method has potential to work well but there is always a trade off with any method. The details are discussed in the next chapter.

3.5 Conveyor Configurations

From section 3.4 it is evident that the Q-frames have a fairly simple assembly process. Although the process is simple, there is always more than one way to accomplish what needs to be done. Using the system modules in different ways lead to totally different reconfigurable manufacturing systems with their own unique characteristics.

The first system module that can be altered is the conveyor layout. The layout of the initial conveyor configuration, set up by the Department of Mechanical and Mechatronic Engineering, cannot be altered because there is not any extra conveyor modules due to financial reasons. However the main task of a simulation analyst is to answer what if questions without needing to physically build a conveyor configuration. Hence three alternative conveyor layouts were designed and modeled in Simio.

Changing the conveyor layout has an impact on the physical appearance of the system, but the system's operating logic can also be changed. Figure 3.11 illustrates the different options that were tested on each of the conveyor configurations.

The first operating logic parameter that is changed is the inspection method. As explained in section 3.4.2 there are two ways in which inspection can be done. In Figure 3.11 *Inspect 1* means that the pallet must visit the inspection & removal station after part placement before welding can continue. *Inspect 2* means that inspection is done at the part feeders and the pallet can continue to the welding

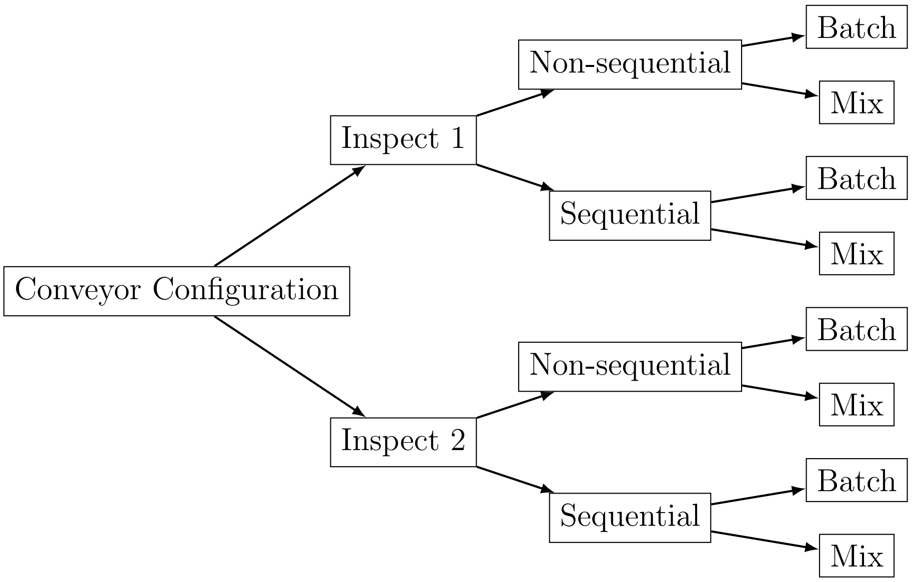


Figure 3.11: General conveyor configuration variations.

station directly after part placement. In both inspection methods, the pallets need to visit the inspection & removal station after welding to remove the parts.

The second operating logic parameter that is changed is the sequence in which parts are placed on the pallet. Section 3.4.1 stated that the parts can be placed in a specific sequence or in any order. In Figure 3.11 *sequential* refers to the process where the pallet has to visit a part feeder type I before it can visit a part feeder type II. *Non-sequential* means that the pallet can visit the two types of part feeders in any order, as long as both are visited before welding.

The final operating logic parameter that is changed is whether all five Q-frames are assembled at the same time or whether it is done in batches. In Figure 3.11 *Batch* refers to when only one type of Q-frame is assembled at a time while *Mix* refers to when all five different Q-frames are assembled at the same time. The process of each of these methods were discussed in section 3.4.3.

3.5.1 Initial Conveyor Configuration (CC1)

The initial conveyor configuration is the physical one that is currently set up in a laboratory, and this system has some limitations. A reconfigurable manufacturing system requires machines with customized flexibility as well as a system that can change capacity quickly. The initial conveyor configuration, referred to as conveyor configuration 1 (CC1), can be used to test the machinery, but as shown in Figure 3.12, the layout of the initial conveyor configuration does not allow for extra capacity. The abbreviations used in this conveyor configuration are repeated here for convenience:

- PM - Pallet Magazine
- PF1 - Part Feeder Type I
- PF2 - Part Feeder Type II
- IR - Inspection & Removal
- W - Welder

The parallel path where the two part feeders are situated causes the system to be unable to change the sequence in which the parts are placed. The inspection method and the use of batches or mixed production schedules can still be tested.

The focus of the simulation study is to understand the characteristics of the system better, rather than focussing on the detailed workings of the machinery. This configuration was modeled as a starting point for learning the simulation software and also to aid the machine designers to better understand the workings of the system.

From Figure 3.12 it is clear that the pallets circulate in a counter clockwise direction. When a pallet is empty and requires parts, it will try to enter the parallel path where the part feeders are situated. If the parallel path is full, the pallet will circulate and try again when it reaches the entrance of the parallel path. The parallel path has a fixed capacity and each machine has a buffer space of one. This means that there can only be a total of four pallets (two receiving service and two waiting for service) in the parallel path at any given time.

Note also in Figure 3.12 that there are transverse conveyor modules at the welding station and at the pallet magazine. The transverse conveyor modules improve the flow of work in the system because the other pallets that are circulating do not have to stop if a pallet is receiving service from that particular station.

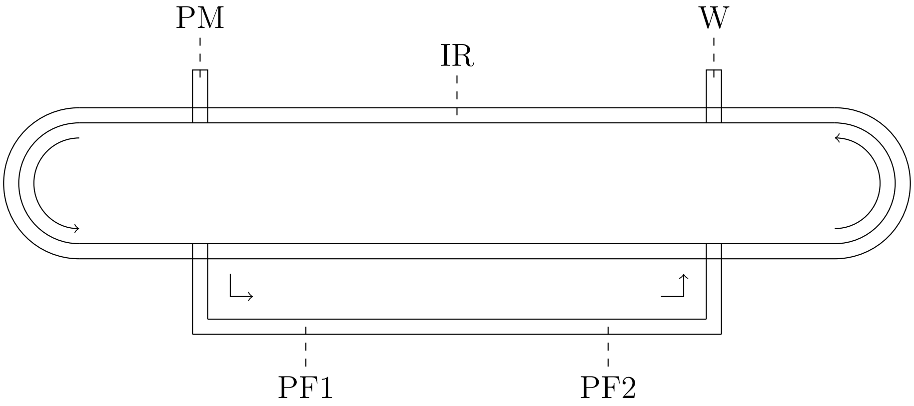


Figure 3.12: Conveyor Configuration 1.

The inspection & removal station does inspection on the production line, which means that other pallets have to stop when the inspection & removal station is busy. The two part feeders operate in a similar way which is why there is only a buffer size of one before each part feeder.

Figure 3.13 shows the Simio model of conveyor configuration 1 for some instance in time when the model is running. At this particular instance, a pallet is just leaving the welding station and another pallet is receiving service from the inspection & removal station. A pallet is also leaving the parallel path after receiving its parts to join the other pallets that are circulating to receive service from the inspection & removal and welding station respectively. Note also that the system is currently running with mixed pallets; each pallet color representing a different Q-frame type as follows:

- Q1 - Red
- Q2 - Blue
- Q3 - Green
- Q4 - Yellow
- Q5 - Orange

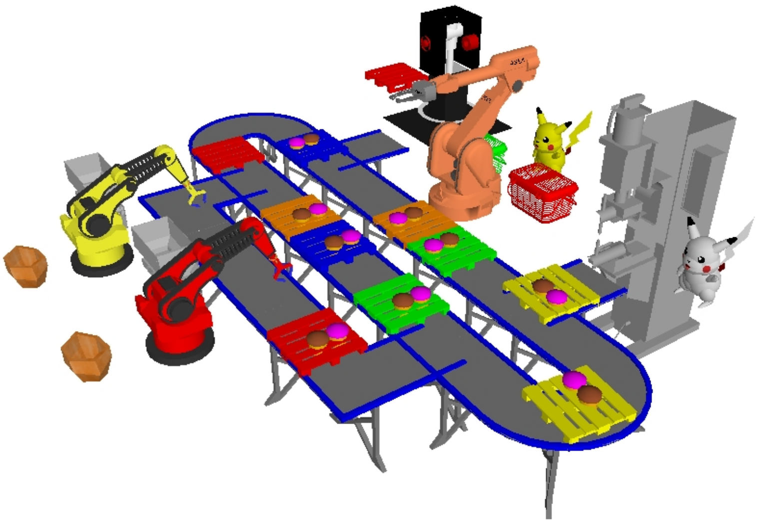


Figure 3.13: Simio model of Conveyor Configuration 1.

3.5.2 Conveyor Configuration 2 (CC2)

Conveyor configuration 2 was the first design attempt that was accepted by all the parties involved in the project. Using the initial conveyor configuration as a starting point, three additional designs were made. Since CC1 had some limitations for reconfigurability, the new designs all had additional capacity as well as different conveyor layouts.

Figure 3.14 shows the layout of CC2 with new abbreviations:

- PM1, PM2 - Pallet Magazine
- PF1, PF2, PF3 - Part Feeder Type I
- PF4, PF5, PF6 - Part Feeder Type II
- IR1, IR2, IR3 - Inspection & Removal
- W1, W2, W3 - Welder

These abbreviations are the same for all the additional designs that were made. An obvious improvement to the initial conveyor configuration is that this system has space for extra capacity. There are now two pallet magazines, six part feeders with three units for each type, three inspection & removal stations, and three welders. This does not mean that the system has to operate at maximum capacity, but for a reconfigurable system to work, it must be designed for reconfigurability from the start. Having a conveyor layout that allows for adding and removing capacity makes this system more reconfigurable than CC1.

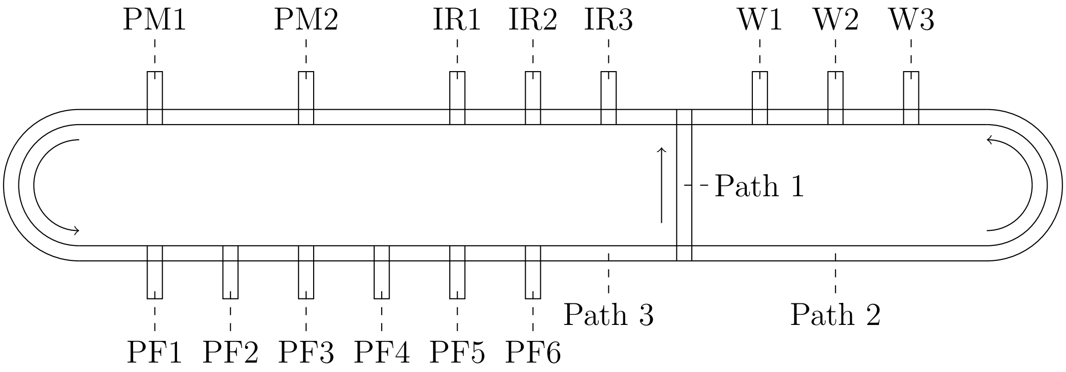


Figure 3.14: Conveyor configuration 2.

CC2 uses transverse conveyors that removes a pallet from the main conveyor loop (Path 2 and 3) when it needs to receive service from a particular station. These transverse conveyors improve the overall flow of the system and it simplifies adding and removing capacity.

As seen in Figure 3.14 path 1 separates the inspection & removal and part feeding stations from the welding stations. Path 1 plays an important role because it effectively shortens a pallet's travel distance. CC2 has a maximum capacity of 36 pallets which leads to pallets competing for the use of the system resources i.e. the machines. All the machines operate in a FIFO (First In First Out) discipline. Since the pallet has to receive all its parts before continuing to welding it may have to circle around more than once because of busy part feeders and inspection & removal stations. The pallet will use the shorter route i.e. path 1 and path 3 instead of circling past the welding stations.

The second use of path 1 is that it creates a large buffer for the welding stations. Only pallets that require welding may enter path 2 which serves as a large buffer for the welding stations. There is a stop gate before the welding stations where pallets wait if the welders are busy. Since the welding stations are the most expensive of all the machinery, it was desired to make the welding stations the bottleneck of the system. With a larger buffer and the correct operating times for the other machines in the system, the welders will be utilized as much as possible, which makes them the bottleneck.

Figure 3.15 shows the Simio model of CC2 at an instance in time when the model is running in batch mode. It clearly shows how the pallets wait in the buffer for the welding stations.

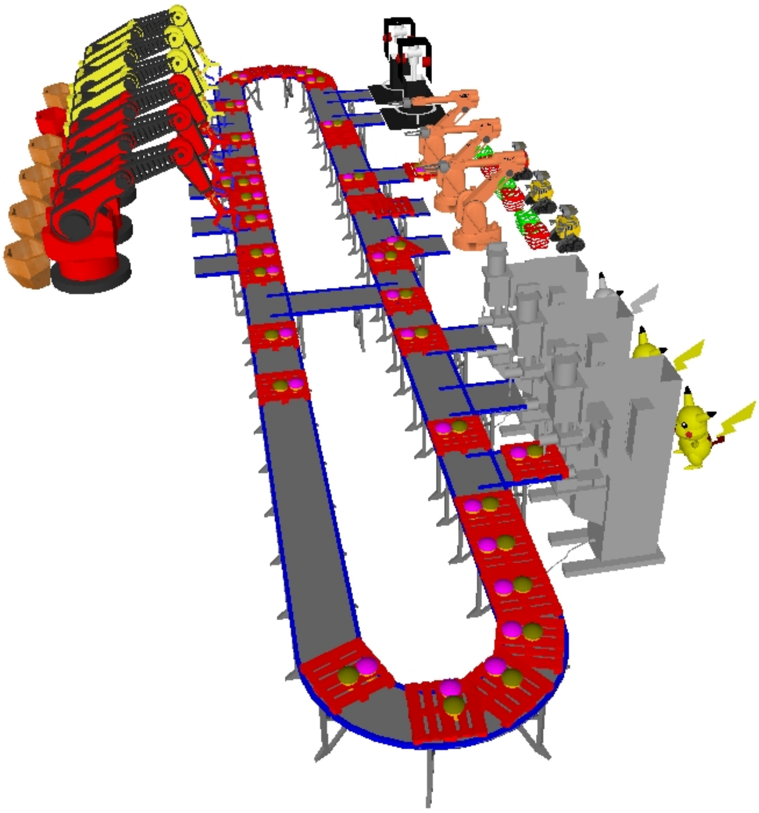


Figure 3.15: Simio model of Conveyor Configuration 2.

3.5.3 Conveyor Configuration 3 (CC3)

After CC2 was developed, it was decided to design two alternatives with the same work station capacity in order to compare the systems to each other. Hence the system module that was altered the most, was the conveyor layout . The location of the machines were also changed to study the effect it had on the system.

Figure 3.16 illustrates the layout for CC3. In this configuration the machines are grouped together to form working cells consisting of two part feeders, an inspection & removal station, and a welder. The pallets are however still free to visit any machine that is available. In this sense it is not a cellular manufacturing system but rather an experiment to see how the system reacts if the machines are organized in this way. Notice that each of the welders have their own parallel path that acts as a buffer with enough space for three pallets. If a pallet requires service from a welder but the buffer of the welder is full, the pallet will keep circulating and try the next welder. Once again transverse conveyors were used at all the machines except the welders because it improves the flow of the system.

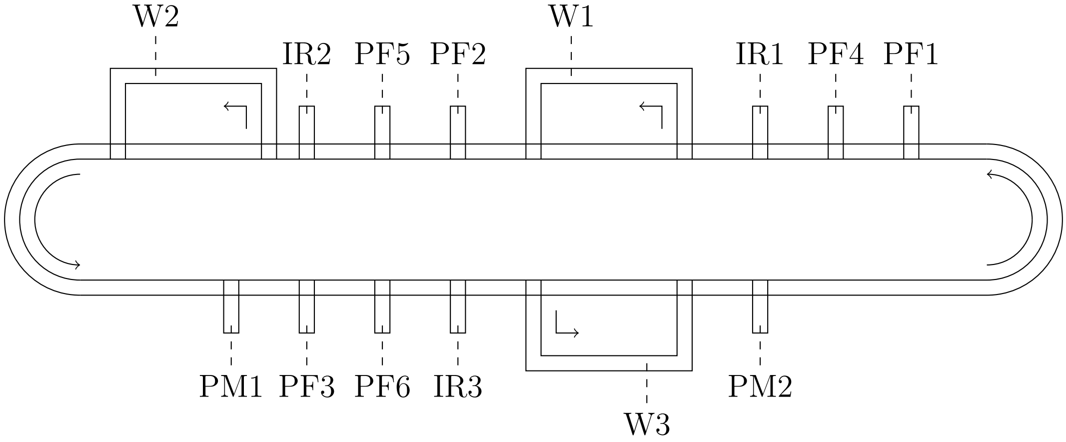


Figure 3.16: Conveyor Configuration 3.

Figure 3.17 shows the Simio model of CC3 at some instance in time when the model is running in mixed mode.

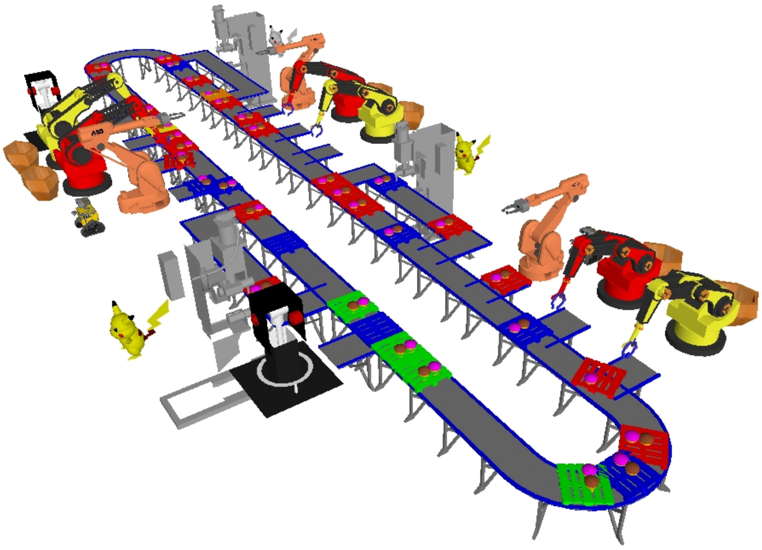


Figure 3.17: Simio model of Conveyor Configuration 3.

3.5.4 Conveyor Configuration 4 (CC4)

CC4 used concepts from each of CC2 and CC3 for its design which is evident in Figure 3.18. CC4 has a large buffer area before all three welders and the machines are organized in cells. This was done to study the effects of using concepts from each of the previous designs and see if it improves the system. Figure 3.19 shows the Simio model of CC4 at some instance in time when the model is running in mixed mode.

An important process that must be followed while designing alternative models, is to verify and validate the models. The verification and validation process for the models developed in this section is discussed in the following section.

3.6 Verification and Validation

Verification and validation are two steps in a simulation study that go hand in hand. Validation is the process of confirming that the model is an adequate representation of the real world system. Verification on the other hand is the process of ensuring that the model is built correctly in the computer program

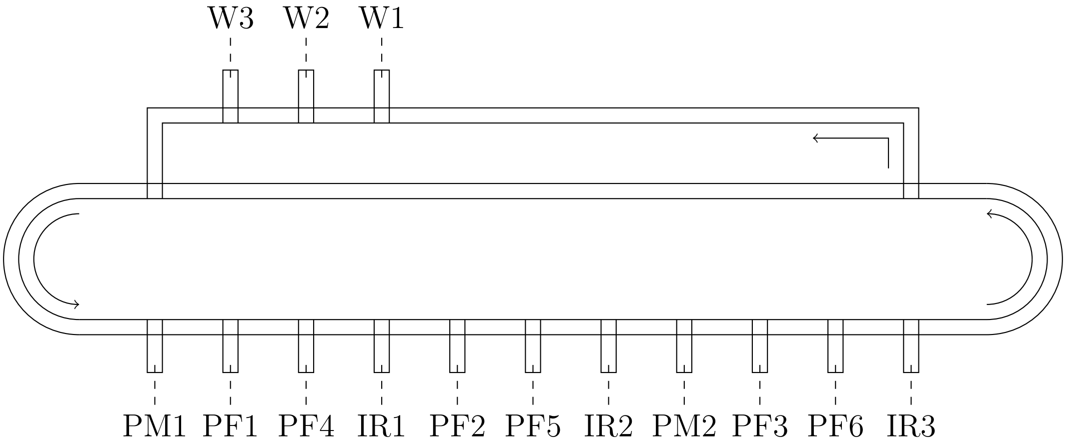


Figure 3.18: Conveyor configuration 4.

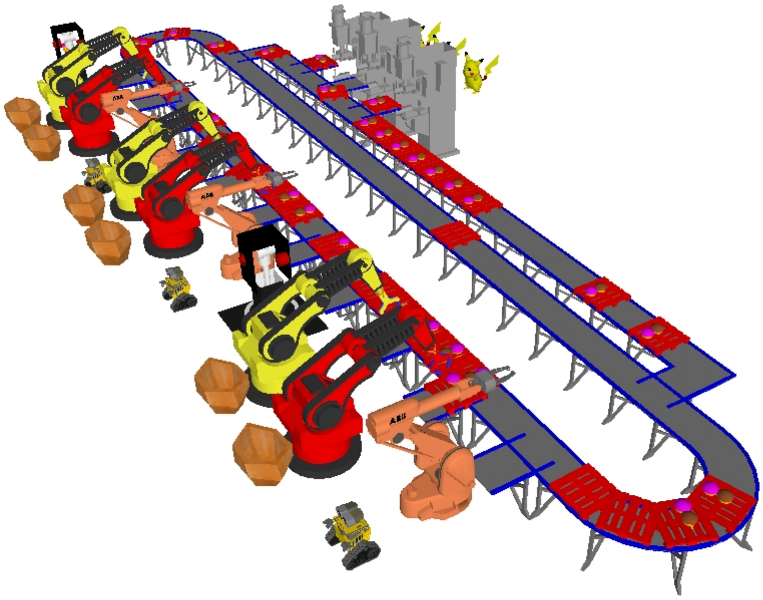


Figure 3.19: Simio model of Conveyor Configuration 4.

that was used to construct the model. In simple terms, verification is debugging the simulation model (Bekker, 2009).

Law (2007) proposes some guidelines to follow in order to verify and validate a simulation model. The goal of verifying and validating a simulation model is so that the model can earn credibility. A model is credible when the people that use the model to make decisions (management), are satisfied with the results of the model and are satisfied with its results. The guidelines followed in this research are not explicitly stated but were applied and will be explained. For further reading on verification and validation see Banks (1998) and Law (2007).

3.6.1 Validating the model

When developing a simulation model, it is important that there is good communication between all the parties involved. The analyst must ensure that the needs of the client are fully understood and this is only possible with regular meetings. For this research project the client was the research project leader from the Department of Mechanical and Mechatronic Engineering. The simulation models and results are supposed to help him and the research team as machine designers to better understand the system. Frequent meetings were held at the start of the project in order to find out exactly what is required from the simulation model.

The initial conveyor configuration is a real world system that could be used to test the validity of its simulation model. Unfortunately the real world system was not up and running by the time that this research was completed. However, meetings were held with the person responsible for getting CC1 operational to gather information on how he expects the system will operate. The dimensions of the conveyor modules were measured in order to build a simulation model of the correct size. Information was also gathered on how the pallets are routed in the system and where they will be stopped with the use of stop gates.

Many of the logic operations that happen in the real world system must be translated into delay times in the simulation model. An example is when a pallet is receiving service from a machine. The first step is to lift the pallet from the conveyor using a pneumatic lift pad. The machine then does the required operation after which the pallet is lowered again onto the conveyor so that it can continue to the next operation. A logic operation like this is modeled as a seize, delay, release process in the simulation model. The seize step represents the lifting of the pallet, the delay is the machine operation and the release step represents the lowering of the pallet. All these steps are assigned a time in the simulation model, so all that is seen is the pallet being stationary at the machine for a period of time. Modeling logical operations like this correctly is a good example of validating the model.

Since none of the models that were developed could be evaluated against existing real world systems, other tests were done to ensure that the output generated from the models were reasonable. The tests as suggested by Law (2007) are as follows:

- *Continuity*: When small changes are made to the input parameters, consequent small and appropriate changes should be reflected by the model's output and variables.
- *Consistency*: Similar runs of the model should result in similar output. Changing the random number stream used for input parameters should not have a large influence on the output.
- *Degeneracy*: The model should reflect the removal of one or more features of the model. Adding or removing capacity in a system should increase or decrease the throughput rate. Machine modules were added and removed from the models and the output reacted accordingly.
- *Absurd Conditions*: If absurd conditions are introduced, the model should not necessarily produce equally absurd outputs. Variables should be in the range specified by the analyst, for example negative mass or negative times are not permissible. The models were tested with absurd conditions, but with modern software, the user is well protected against absurdities.

3.6.2 Verifying the model

As previously stated, the verification process is basically debugging the simulation model. Simulation models are like any computer program, meaning that there is more than one way to code a program in order to achieve the same results. No two persons will build the same model exactly the same. It is therefore vital to get a knowledgeable person's opinion about the model during development. Techniques suggested by Law (2007) were used during the model building phase and they are the following:

- The model was developed using modules and subprograms. After a module is completed, it is debugged because it is poor programming practice to develop the entire model before debugging it. A large model has too many variables and determining the location of errors is more difficult.
- The simulation model was run under a variety of settings to check if the model's output is still reasonable.
- A structured walk-through with the research supervisor was held regularly to confirm that he agreed with the way the model is programmed.
- Simio has the model trace function that allows the user to trace the state of the system, i.e., the state variables, certain statistical counters, movement of entities etc. All this information is displayed after each event in the system and can be compared to hand calculations. The trace function was used to ensure that the entities react as required.
- The model was run under simplified conditions with deterministic input parameters. It is then possible the calculate by hand what the output should be and verify that the model is working correctly.

3.7 Chapter summary

This chapter focussed on the parts that need to be assembled as well as the system modules that are used to do the assembly. Organizing these system modules into different configurations gave rise to four different simulation models that can be used to study the characteristics of reconfigurable manufacturing systems. During the development of the different configurations, the verification and validation techniques discussed in sections 3.6.2 and 3.6.1, were applied. In the next chapter the experiments conducted are outlined and explained in more detail.

CHAPTER *4*

DESIGN OF EXPERIMENTS

Chapter 3 explained the different conveyor configurations developed. Simulation models of these configurations are used to generate output data that can be used to make meaningful recommendations. This chapter will focus on the experiments that were conducted in order to generate the required output data.

The controls used to drive the experiments as well as the the measured responses will be discussed. With all experiments, some assumptions are made which must be justified by explanation. The type of experiments that were made including validation, Pareto exhaustive searching, conveyor speed adjustment, and ramp-up time estimation are discussed.

4.1 Experiment Controls

Simulation models use input data which is usually assigned to variables to drive the model through time. Events in the system during a run change the values of variables and these variables are then used to calculate certain responses of the system. By changing the input data that is assigned to the variables, the analyst is capable of testing different operating scenarios of the system. Testing different scenarios makes it possible to answer many what if questions and enables the analyst to determine the characteristics of the system.

For each of the conveyor configurations that was designed, it was decided that the layout of each particular configuration is fixed. In chapter 3, section 3.5 it was explained that although the conveyor layout is fixed, the operating logic of the system can still be changed. Recall from Figure 3.11 that the operating logic parameters that can be changed are the inspection method, the sequence of part placement, and batched or mixed pallets. Figure 3.11 is repeated here in Figure 4.1 for convenience.

For the simulation model to work correctly in Simio, each type of operating logic had to be hard programmed into the logic of the model. From Figure 4.1 it can be seen that each configuration has eight variations based on operating logic. The initial conveyor configuration (CC1) is unable to change the sequence of part placement as explained in section 3.5.1. Hence the overview for CC1 differs from the other configurations and can be seen in Figure 4.2. Note that CC1 has only four variations based on operating logic. In total there are 28 Simio models that were developed and tested with experiments. Some of the models were discarded after initial experiments which will be explained in a subsequent section.

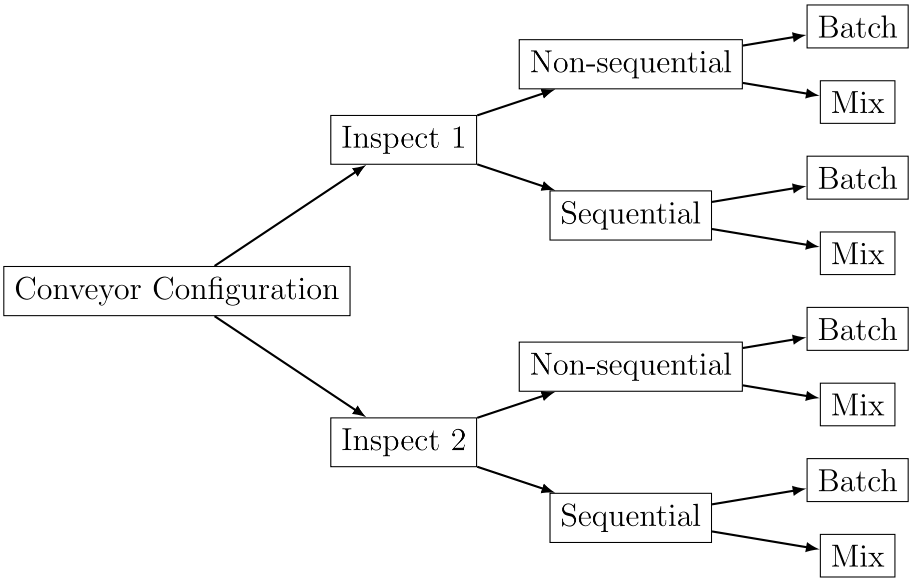


Figure 4.1: General conveyor configuration variations.

Once a combination of operating logic parameters has been chosen, which is one out of 28 possibilities, the actual experiment can be performed. The first step is to choose and set values for the controls of the model. The following controls for experimentation were chosen:

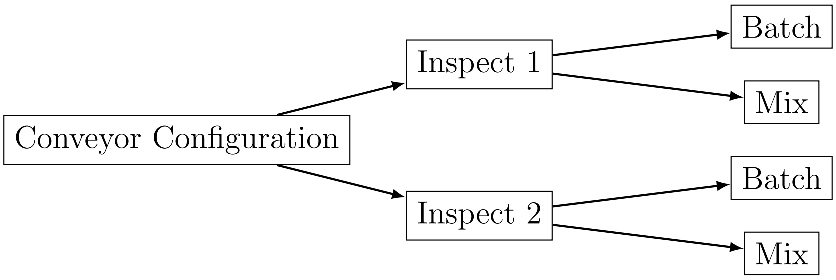


Figure 4.2: Initial conveyor configuration variations.

- *System Capacity*: This is the combination of machines that will be used during the simulation run and can be specified in two ways. The first way is a combination of machines that will remain unchanged during the entire run. The second way is a initial machine combination that will change during a run as the system undergoes reconfiguration.

- *Order*: The order is an Excel sheet containing the orders for each of the five Q-frames. The order for each Q-frame can be changed to any non-negative value. The size of batches can also be specified.

- *Pallets*: The number of pallets assigned to each type of Q-frame is calculated as a proportion of the total number of pallets based on the pallet's relative order quantity. When the system is running in batch mode, all the Q-frames are assigned the same number of pallets.

- *Machine processing times*: These are the times the machines take to perform their operations. The processing times are modeled as triangular distributions.

- *Conveyor Speed*: The speed of the conveyor can be set to different values in meters per minute (m/min). Once the speed is set, it will stay the same for an entire simulation run.

- *Batch delays*: It is the time it takes to change over from one batch to another.

- *Reconfiguration delays*: It is the time it takes to add and remove capacity from the system. Each type of machine takes a different time to add and remove and these times are modeled as triangular distributions.

- *Maintenance*: It is common that machines have count based maintenance performed on them during production. The most affected machine is the welding station. The electrodes of the welder should be cleaned after a specified number of pallets have been welded. The welder will be unavailable for service for a specified time which simulates the cleaning of the electrodes.

4.2 Experiment Responses

Like most modern simulation software, Simio generates a vast amount of output data. The data is organized in columns according to the scenario number, object type, object name, data source, category, data item, and statistic type. A scenario represents a single set of input data that is tested on the model. An experiment usually consists of more than one scenario, in order to evaluate different options.

The object type is one of the system modules like an inspection & removal station or a conveyor module and the object name is the specific name of that module in the system. The data source is the part of an object that generated a specific data entry, for example the contents of what a part feeder has processed. The data generated by the different data sources of the objects are then further sorted into categories, type of data item and statistic type. The last three columns exist to make it easier to find a specific data entry.

Although Simio collects large amounts of data, the user can still let Simio collect user defined data as well. User defined output will usually use the user defined variables to calculate the required output. The user defined output statistics of interest are as follows:

- *Sum of Ramp-up Times*: The system starts production from an empty state and it takes time before the first correctly assembled Q-frame is produced. The time from startup until the first correctly assembled Q-frame is calculated as a ramp-up time. When the system is changing between batches, all the pallets must first be removed from the system before the new batch of pallets can start production. There is usually a delay between batches because the pallet magazines need to be restocked with new pallet cartridges. The time from when production stops to remove pallets until the first Q-frame of the next batch is assembled correctly, is also a ramp-up time. The final ramp-up time is calculated when the system is being reconfigured. Ramp-up time is from the moment production stops until reconfiguration is completed and the first correctly assembled Q-frame is produced. *Sum of ramp-up times* is a statistic that takes the sum of all three types of ramp-up times that occur during a run.

- *Number of Defects*: The number of defective units produced for each type of Q-frame is calculated. Defects occur according to a certain probability which results in Q-frames with higher production quantities, having more defects produced.

- *Number of Good*: The number of correctly assembled units produced for each type of Q-frame is calculated. This value must always equal the order size for a particular Q-frame at the end of a run. This statistic is also used as a stopping condition for the model. Once all the orders have been filled, the model stops running.

- *Throughput Time*: This is the time it took the system to produce a required order. The individual Q-frame throughput times are calculated as well as an overall system throughput time. Throughput time is calculated in hours.

- *Throughput Rate*: This is one of the most important output measures of the system. The throughput rate of the system is the number of good units it can produce per hour for an entire run. It includes all the delays that the system may experience during a run. The throughput rate for each of the Q-frames are also calculated, but the main measure is still the throughput rate of the system.

- *Work in Progress* (WIP): Once a pallet has received all its parts and it is circling the conveyor system awaiting service, the pallet is seen as work in progress. It is desirable to keep WIP as low as possible because it represents tied up capital.

4.3 Assumptions

The scope of this study is focussed only on how the configurations operate under ideal conditions. Errors that are external to the configurations, but have an effect on the configurations were not considered. Hence in order to build the models and do experiments, the following assumptions were made.

- Machine breakdowns were not considered. The reason is that breakdowns occur randomly in systems which makes it difficult to compare two or more configurations to each other if one of the configurations has a breakdown during a run. The configuration with the breakdown is obviously at a disadvantage. However, the count based maintenance performed on the welders can be seen as a "breakdown". It is uniform across all the configurations, so none of them is at a disadvantage.

- Power failures that will cause a momentary lapse in production were not considered.

- If a particular configuration is chosen and it is actually built, the assumption is made that there will be more than one of these configurations on the factory floor. The reason for this is to justify removing capacity from a single configuration. Removing capacity from a system and letting a machine stand idle does not make any sense. Hence if a machine is removed from a configuration, it is assumed that that machine is added to another configuration. The result is a factory floor of configurations which utilize machine modules from a pool of machine resources. A configuration is reconfigured with just enough capacity to meet its orders.

- There is sufficient personnel to carry out the reconfigurations when necessary.

- There is never a material shortage or not enough parts to fill a order.

- Once a system is reconfigured and machines are added or removed from the system, there is not a time period for calibrating the system. Calibration time is included in the reconfiguration time. The system is assumed to be operating as normal only with less or more machines.

- All the pallets are removed from the system between batches and when reconfiguration takes place.

- When the system is running in mixed mode and the order for a particular Q-frame is completed, that Q-frame's pallets are removed from the system. For example the order for Q5 is completed before the other Q-frames, the Q5 pallets are removed to prevent excess Q5's from being produced and wasting system resources.

- The welding station uses different tool heads to weld the different Q-frames. The implication of this assumption is that each pallet is assigned two welding times. Welding time 1 is the time spent at the welding station when the welder does not have to change a tool head. Welding time 2 is for when the welder does have to change it. Suppose pallets arrive at the welder in the sequence Q1, Q1, Q2, Q3, Q3. Not knowing which tool head the welder has, the first Q1 will spend welding time 2 at the welder. The second Q1 will spend welding time 1 at the welder, since the welder has the correct tool head. Q2 and the Q3 will spend welding time 2 and the second Q3 will spend welding time 1. For the count based maintenance performed on the welder to clean the electrodes, it is assumed that the welder will be unavailable for 20 seconds.

The experiment controls discussed in section 4.1 create different experiments through changing their values. The one set of controls that do not change across all the experiments are the machine processing times. Since many of these machines have not been developed by the time this research was completed, the processing times of the machines were an educated guess. Meetings were held with the machine designers from the Department of Mechanical and Mechatronic Engineering to determine what times they expect the machines will be operating at. The one assumption is that the welding station is the bottleneck, hence its processing time will be the longest. Since the welder has to complete five welds for each pallet, this assumption seemed reasonable. All the machines except the pallet magazine operate on a seize, delay, release principle as explained in the example in section 3.6.1. The machine processing times can be seen in Table 4.1 with all times in seconds. In Table 4.1 *Triangular* refers to the statistical triangular distribution with minimum, mode, and maximum as parameters.

Table 4.1: Machine Processing Times (seconds).

Machine Type	**Seize**	**Delay**	**Release**
Part Feeder Type I	0.4	Triangular(1, 1.1, 1.2)	0.4
Part Feeder Type II	0.4	Triangular(2.8, 3.2, 3.5)	0.4
Welding Time 1	0.4	Triangular(8.5, 9, 9.2)	0.4
Welding Time 2	0.4	Triangular(13.5, 14, 14.2)	0.4
Inspection & Removal	0.4	Triangular(1.5, 1.8, 1.9)	0.4
Pallet Magazine	0.0	Triangular(15, 20, 30)	0.0

The actual times that are used for the model is not all that important. The important thing is that the output generated by the model should be interpreted in terms of the input data of the model. With the model already built in Simio, it is easy to change the input data once the real operating times are known. *Experiments using different input data than was used in this research can be conducted in the future by other researchers.* For the purpose of this research it was desired to evaluate the differences of conveyor layouts and operating logics. Changing the machine processing times add to much variability to make a comparison. As already stated it is important to interpret the results in terms of the input data for the model.

The final assumption that was made is the probability of assembling a defective part. The probability was chosen as 0.02. This means that every time a pallet enters an inspection & removal station, there is a 0.02 chance that the part is defective after which it is removed from the pallet and discarded or reworked.

4.4 Validation Experiment

The aim of the validation experiment was to establish two facts about the configurations. The first was to test whether the models reacted correctly when capacity was added and removed from the systems. The second was to statistically compare the models to each other to ensure that they are different. Two models can look different but the output that they deliver may not differ significantly enough to be able to choose between the two.

The output statistic that was used as a measure for this experiment is the throughput rate of the system in units per hour. Changing the capacity of the system should be reflected in the throughput rate of the system. The throughput rate was also used to compare the configurations to each other. CC1 was not used in this experiment, since it is not possible to change its capacity.

The configurations were given an initial capacity that remained unchanged during a run and each run was replicated ten times. Ten replications were deemed sufficient because the half width of the throughput rate was small and because a large number of experiments had to be done. Typical half widths are shown in Table 4.2.

Table 4.2: Typical Half Widths

Conveyor Configuration	**Mean Throughput Rate**	**Half Width**
CC2	229.05	0.98
CC3	227.87	1.09
CC4	215.09	1.09

Note that the pallet magazines were never removed, so there are always two for CC2, CC3, and CC4. The capacity combinations that were tested are as follows (Abbreviations are given as they appear in the Excel output report):

- *MaxCap*: Maximum capacity which is three part feeders each of type I and II (six in total), three inspection & removal stations, and three welders.
- *MinCap*: Minimum capacity which is one part feeder each of type I and II, one inspection & removal station, and one welder.

- *2 of each*: Two of each type of machine.
- *2PF,2IR,1W*: Two part feeders each of type I and II, two inspection & removal stations, and one welder.
- *3PF,3IR,1W*: Three part feeders each of type I and II, three inspection & removal stations, and one welder.
- *3PF,3IR,2W*: Three part feeders each of type I and II, three inspection & removal stations, and two welders.

A standard order had to be assembled by all the configurations across all capacity combinations. The order for each frame as well as the number of pallets assigned to each frame can be seen in Table 4.3. Note the large variation of the order size across the different frames. The experiments were conducted with 36 pallets on the conveyor at all times, and a conveyor speed of six meters per minute. When a configuration was in batch mode, there would be 36 pallets of the same type of Q-frame on the conveyor at a time. When a configuration was in mixed mode, the pallets are assigned to different Q-frames proportionally as shown in Table 4.3.

Table 4.3: Order size and associated pallets.

Q-frame	**Order size**	**Number of Pallets**
Q1	5866	21
Q2	2533	9
Q3	1200	4
Q4	267	1
Q5	134	1
Total	10000	36

4.5 Pareto Exhaustive Searching Experiment

In the validation experiment, only one experiment response i.e. system throughput rate was considered. When trying to optimize a system, it is good practice to not only focus on one response because there is always a price to pay for reaching a certain level of performance. Therefore it was decided to do multi-objective optimization on the configurations, so that a more informed decision can be made when choosing a configuration.

For multi-objective optimization, two or more contradicting objectives are chosen that will result in a trade off curve called the *Pareto front*. Two objectives are contradicting if the one objective can only be improved at the cost of the other objective. For most manufacturing applications, it is desired to make as many products as possible for the least amount of money while still maintaining a high level of quality.

For the conveyor configurations under question, making as many products as possible, means maximizing the throughput rate of the system. However, a high throughput rate comes at a price. The number of machines and pallets used to assemble the Q-frames can be seen as the cost to the system. An experiment response that is a representation of the cost to the system was required. The average work in progress (WIP) is the number of pallets that have received all their parts, and it was chosen because it will increase or decrease if the number of machines and pallets in the system increase or decrease. Note that the average WIP is not equal to the total number of pallets that are placed on the conveyor, but rather the number of pallets that have received their parts and are awaiting service. Therefor an empty pallet is not considered as WIP and WIP is measured in number of pallets. The pallets used in the configurations are equipped with RFID tags and are made of heavy, durable materials. This causes the pallets to be expensive and it is desired to determine the least number of pallets required for the highest throughput rate.

The number of machines used to produce a certain throughput rate must also be considered because it is a cost to the system. However, if a capacity combination is chosen it will stay fixed for an entire batch or production order. Therefore the only **variable** that can influence the cost to the system is the number of pallets on the system. The number of pallets in the system has a direct influence on the WIP that is measured as an output. A cost analysis is beyond the scope of this project since accurate cost estimations are unavailable. The reason being that many of the machines have not been designed yet.

In conclusion, the aim of the Pareto exhaustive searching experiment is to **maximize** the throughput rate of the system while at the same time **minimizing** the average work in progress.

After the validation experiment, an important discovery was made that affected the remainder of the experiments. It was found that the sequence in which the parts are placed does not affect the output of the system. Either way, the system delivers the exact same results. The reason for this is that once all the pallets are on the system, the system reaches equilibrium after a short period of time. The pallets will enter any machine that is available, and since the conveyor is circular, the sequence in which parts are placed becomes irrelevant. For this reason, the general overview of the conveyor configurations was revised and is shown in Figure 4.3. Each model now has only four variations instead of eight which results in 16 models in total instead of 28.

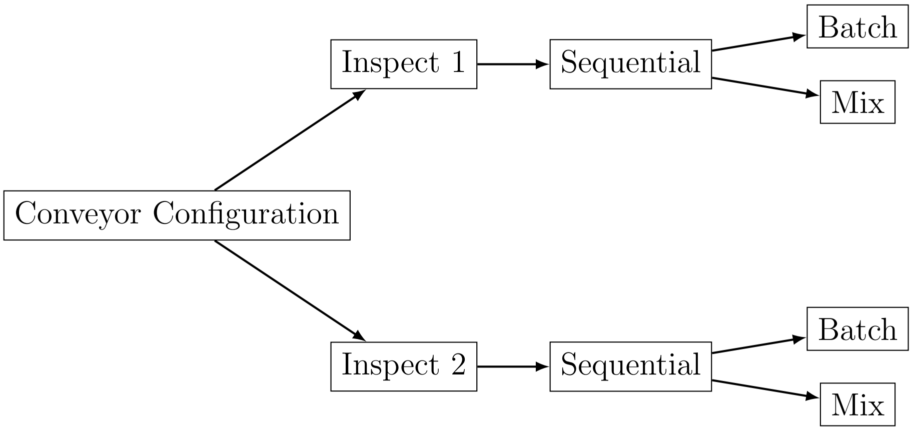


Figure 4.3: Revised conveyor configurations variations.

To find a Pareto front, the objective space must first be determined. Using cartesian coordinates, the objective space will have throughput rate on the vertical axis and WIP on the horizontal axis with the objective to maximize the former and minimize the latter. One point in the objective space is a combination of the throughput rate and WIP for a single experiment scenario. The scenarios are chosen by changing the experiment controls to develop different what if scenarios.

The experiment controls that were changed during the Pareto exhaustive searching experiment are the following:

- Capacity of the system.
- Total number of pallets.
- Conveyor speed.

Like in the validation experiment, a capacity combination was assigned to the configuration that remained unchanged during the run. The total number of pallets in the system was varied between 10 and 36 in increments of one. The number of pallets assigned to each type of Q-frame during batch mode was equal to the total number of pallets proposed. The number of pallets assigned to each type of Q-frame during mixed mode can be seen in Table 4.4.

The total number of pallets were varied as shown in Table 4.4 for three different conveyor speeds which included 6, 9 and 18 meters per minute. With 27 different pallet combinations, and three different conveyor speeds, there are 81 scenarios for each capacity combination. CC1 however is a smaller system and does not have the capacity for 36 pallets. The total number of pallets were varied between five and 12 and can be seen in Table 4.5. The conveyor speeds were varied like in all the other configurations which results in a total of 24 scenarios. The final objective space consisted of 5928 different scenarios from which the Pareto front is determined. The same orders used in the validation experiment were used in the Pareto exhaustive searching experiment and can be seen in Table 4.3.

4.6 Conveyor Speed Adjustment Experiment

During the Pareto exhaustive searching experiment, it was noted that when the configurations were running in mixed mode, the same output was delivered for the same number of pallets at different conveyor speeds. It seemed that the output of the configuration reached a ceiling value. This phenomenon did not appear when the configurations were running in batch mode. It was decided to test a selection of the mixed configurations with varying conveyor speeds to try and determine at which conveyor speed the system's output reaches the ceiling value. Since this phenomenon was discovered while the configurations used conveyor speeds of 6,9,

Table 4.4: Pallet variation for Pareto exhaustive experiment.

Q1 Pallets	**Q2 Pallets**	**Q3 Pallets**	**Q4 Pallets**	**Q5 Pallets**	**Total Pallets**
21	9	4	1	1	36
20	9	4	1	1	35
20	8	4	1	1	34
19	8	4	1	1	33
19	8	3	1	1	32
18	8	3	1	1	31
17	8	3	1	1	30
17	7	3	1	1	29
16	7	3	1	1	28
15	7	3	1	1	27
15	6	3	1	1	26
14	6	3	1	1	25
14	6	2	1	1	24
13	6	2	1	1	23
12	6	2	1	1	22
12	5	2	1	1	21
11	5	2	1	1	20
11	4	2	1	1	19
10	4	2	1	1	18
9	4	2	1	1	17
9	4	1	1	1	16
8	4	1	1	1	15
8	3	1	1	1	14
7	3	1	1	1	13
7	2	1	1	1	12
6	2	1	1	1	11
5	2	1	1	1	10

and 18 meters per minute it was obvious that lower speeds needed to be tested and not higher speeds.

Table 4.5: Pallet variation for CC1.

Q1 Pallets	Q2 Pallets	Q3 Pallets	Q4 Pallets	Q5 Pallets	Total Pallets
6	3	1	1	1	12
5	3	1	1	1	11
4	3	1	1	1	10
4	2	1	1	1	9
3	2	1	1	1	8
3	1	1	1	1	7
2	1	1	1	1	6
1	1	1	1	1	5

An experiment was set up where the total number of pallets were varied between 10 and 30 in increments of five as shown in Table 4.6. Six different conveyor speeds were tested which included 1, 2, 4, 6, 9, and 18 meters per minute. The phenomenon was detected across all capacity combinations, so it was decided to do the conveyor speed adjustment experiment at maximum capacity to save simulation time. A total of 30 different scenarios for each of the chosen configurations was tested.

Table 4.6: Pallet variation for conveyor speed experiment.

Q1 Pallets	Q2 Pallets	Q3 Pallets	Q4 Pallets	Q5 Pallets	Total Pallets
5	2	1	1	1	10
8	4	1	1	1	15
11	5	2	1	1	20
14	6	3	1	1	25
17	8	3	1	1	30

4.7 Ramp-up Time Experiment

For the ramp-up time experiment, a production scenario was created where a configuration undergoes reconfiguration by adding and removing capacity according to the required production levels. Table 4.7 shows the order quantities for each of the different Q-frames, the total number of pallets used for each order, and the capacity the system used to assemble the order. The system will start at minimum capacity and it will reconfigure quickly each time the orders in a row of Table 4.7 are completed. If the system is running in batch mode, the number of pallets for each batch is equal to the *total pallets* in Table 4.7. If the system is running in mixed mode, the pallets are assigned to each type of Q-frame as shown in Table 4.8.

Table 4.7: Production scenario for ramp-up time experiment.

Q1	**Q2**	**Q3**	**Q4**	**Q5**	**Total Pallets**	**Capacity**
500	300	100	30	25	10	MinCap
1000	500	120	33	30	15	2PF,2IR,1W
1500	900	150	50	40	20	2 of each
2200	1300	160	60	55	30	3PF,3IR,2W
450	290	70	25	22	10	MinCap
1300	800	140	35	35	20	2PF,2IR,2W
3000	1600	200	80	75	35	MaxCap
2300	1100	140	31	45	30	3PF,3IR,2W
900	400	100	20	25	15	2PF,2IR,1W
1600	780	130	44	36	20	2 of each

The ramp-up time is dependent on the estimated times to add and remove capacity i.e. the reconfiguration delays shown in Table 4.9. Note that *two part feeders* refer to one of each type of part feeder. The batch delays were estimated as Triangular(8, 10, 12) minutes.

4.8 Chapter Summary

In this chapter the experiments that were designed to asses the characteristics of the conveyor configurations were explained. The experiments that were conducted are the validation, the Pareto exhaustive search, the conveyor speed adjustment, and the ramp-up time experiment. Each experiment was designed to

Table 4.8: Mixed pallets for ramp-up time experiment.

Q1 Pallets	**Q2 Pallets**	**Q3 Pallets**	**Q4 Pallets**	**Q5 Pallets**	**Total Pallets**
5	2	1	1	1	10
8	4	1	1	1	15
11	5	2	1	1	20
17	8	3	1	1	30
5	2	1	1	1	10
11	5	2	1	1	20
20	9	4	1	1	35
17	8	3	1	1	30
8	4	1	1	1	15
11	5	2	1	1	20

Table 4.9: Reconfiguration delays.

System Module	**Installation Time (min)**	**Removal Time (min)**
Inspection & removal	Triangular(15, 20, 25)	Triangular(5, 7, 10)
Two part feeders	Triangular(27, 32, 35)	Triangular(11, 15, 17)
Welder	Triangular(22, 25, 30)	Triangular(8, 10, 13)

test a different characteristic of the developed configurations. In the next chapter the results that were obtained from these experiments are presented and discussed.

CHAPTER 5

RESULTS AND DISCUSSIONS

Chapter 4 focussed on how the experiments were developed and what input data was used to perform the experiments. In this chapter the output or results obtained from the experiments are presented and discussed. The results are presented in the same order in which the experiments were discussed, starting with the validation experiment and ending with the ramp-up time experiment.

The output from simulation models are used to identify trends and give an approximation of how the system reacts under specific operating conditions. Simio records large amounts of data that enables the analyst to identify the required trends. Examples of what was discovered will be shown in this document but for a green initiative aimed at saving paper, the details of all the experiments can be viewed on the CD provided at the back of the document.

5.1 Results of the validation experiment

The validation experiment had two purposes of which the first was to test whether or not the configurations reacted correctly to changes of capacity in the system. After 10 replications for each of the capacity combinations discussed in section 4.4, the average of the throughput rate of the system is calculated. An example of the average throughput rate for CC2, CC3, and CC4 can be seen in Table 5.1. The conveyor configuration variation used to generate the data in Table 5.1

is when the configurations are using inspection method 1 and running in mixed mode.

Table 5.1: Capacity combinations results for CC2, CC3, and CC4.

Capacity	**CC2**	**CC3**	**CC4**
MinCap	85.50	79.26	79.79
2PF,2IR,1W	122.10	154.60	123.01
2 of Each	162.80	152.74	154.33
3PF,3IR,1W	122.18	218.61	122.95
3PF,3IR,2W	207.49	225.23	208.55
MaxCap	227.17	215.43	221.13

One of the assumptions made in section 4.3 is that the welding stations must be the bottleneck because they are the most expensive of all the machines. The machine processing times were chosen in such a way that the welding stations are indeed the bottleneck which is proved in Table 5.1 and Table 5.2.

At minimum capacity the throughput rate of the system is low because the the single welder is starved from the lack of part feeders and inspection & removal stations. Table 5.2 shows the welder utilization of CC2 and note that at minimum capacity, the welder is only utilized 67.48% of the time. When the capacity of the part feeders and inspection & removal stations are doubled while the number of welders are kept at one, the utilization is improved to 96.47% while the throughput rate is increased from 85.50 to 122.10 parts per hour. This trend of the configurations performing better when there are more part feeders and inspection & removal stations, than there are welders, is clearly visible from Table 5.1 and Table 5.2. For the results of the capacity combinations for the other conveyor configuration variations, refer to the CD.

The second purpose of the validation experiment was to ensure that the different configurations do not only differ in conveyor layout, but also to statistically compare their outputs to ensure that the systems differ significantly. The first test that was performed is the paired-t confidence interval technique, as discussed in section 2.8.4 on page 44. Z_1, Z_2, and Z_3 are the new variables created to represent the difference between two systems. For example, Z_1 is the difference in

Table 5.2: Welder utilization of CC2 for different capacity combinations (percentage).

Capacity	**Welder 1**	**Welder 2**	**Welder 3**
MinCap	Not used	Not used	67.48
2PF,2IR,1W	Not used	Not used	96.47
2 of Each	Not used	70.72	74.94
3PF,3IR,1W	Not used	Not used	96.44
3PF,3IR,2W	Not used	93.69	94.07
MaxCap	70.71	76.47	79.71

throughput rate for CC3 and CC2. $\bar{Z}$ is the average of Z_1 since 10 replications were made, resulting in 10 observations of Z_1. $Var(Z)$ is the variance of $\bar{Z}$ and t-value is the critical value of the student-t distribution. Note that because three systems are compared to each other, the Bonferroni inequality was applied to determine the level of confidence. To ensure that a confidence level of at least 90% is reached i.e $\alpha = 0.1$, the confidence level for the individual confidence intervals must be calculated using $1 - \frac{\alpha}{[\frac{k(k-1)}{2}]}$ (Law, 2007). In this case with three systems, the new individual confidence level is 96.67%. The result is that instead of using a t-value of 1.83, a t-value of 2.51 is used. The half width indicates the coverage of the confidence interval and is the deviation from $\bar{Z}$ for the lower and upper bounds. If the confidence interval for $Z_i, i = 1, 2, 3$ does not include 0, the two configurations differ significantly enough to be regarded statistically as two different systems. From Table 5.3 it is clear that the throughput rate of CC2 is greater then the throughput rate of CC3, with 90% confidence, because the confidence interval limits are both negative. CC2 also has a higher throughput rate than CC4, but CC4 has a higher throughput rate than CC3 at maximum capacity. Table 5.4 shows that the systems also statistically differ when running in *inspection method 2*, *batch mode*, and with *2 of each* capacity.

A second test was also performed to confirm the findings of the t-test. Notches were calculated as explained in section 2.8.4.3 on page 46. Recall that a notch is calculated around the median and not the mean. The spread of the output data is better represented with this method. If the notches of the three systems do not overlap, there is statistically significant difference in the output of the respective

Table 5.3: Paired-t confidence interval results for maximum capacity.

Statistic type	$\mathbf{Z_1(n)} = \mathbf{CC3 - CC2}$	$\mathbf{Z_2(n)} = \mathbf{CC4 - CC2}$	$\mathbf{Z_3(n)} = \mathbf{CC4 - CC3}$
$\bar{Z}$	-11.74	-6.04	5.69
$Var(Z)$	1.80	0.60	2.46
t-value	2.51	2.51	2.51
Halfwidth	3.36	1.95	3.94
Lower Bound	-15.10	-7.99	1.75
Upper Bound	-8.37	-4.10	9.63

Table 5.4: Paired-t confidence interval results for 2 of each capacity.

Statistic type	$\mathbf{Z_1(n)} = \mathbf{CC3 - CC2}$	$\mathbf{Z_2(n)} = \mathbf{CC4 - CC2}$	$\mathbf{Z_3(n)} = \mathbf{CC4 - CC3}$
$\bar{Z}$	28.24	-0.93	-29.17
$Var(Z)$	22.69	0.02	23.34
t-value	2.51	2.51	2.51
Halfwidth	11.96	0.33	12.12
Lower Bound	16.29	-1.26	-41.30
Upper Bound	40.20	-0.60	-17.05

systems. Table 5.5 supports the findings of Table 5.3 while Table 5.6 supports the findings of Table 5.4.

Table 5.5: Notch calculations for maximum capacity.

Statistic Type	**CC2**	**CC3**	**CC4**
R	2.40	2.70	2.48
N	10.00	10.00	10.00
Median	229.53	227.36	214.90
Upper Bound of notch	230.50	228.46	216.11
Lower Bound of notch	228.55	226.27	213.69

Table 5.6: Notch calculations for 2 of each capacity.

Statistic Type	**CC2**	**CC3**	**CC4**
R	0.04	6.22	0.07
N	10.00	10.00	10.00
Median	228.26	212.08	240.27
Upper Bound of notch	228.27	214.61	240.30
Lower Bound of notch	228.24	209.56	240.24

5.2 Pareto Exhaustive Search Experiment Results

In the Pareto exhaustive search experiment, a large number of scenarios was developed to ensure that all possibilities are considered within the range of experiment controls. In total, 5928 different scenarios were tested and each scenario was replicated ten times. The experimental responses of interest are the throughput rate (TPR) of the system and the work in progress (WIP). The throughput rate of the system is measured in parts per hour while the WIP represents the average number of pallets that have received all its parts at any given time during the run. Hence WIP is measured in pallets.

The TPR of the system and the WIP constitute the objective space for the Pareto exhaustive search experiment. Figure 5.1 shows the results obtained from all the scenarios with the throughput rate on the vertical axis and the WIP on the horizontal axis.

From the objective space, the Pareto front can be extracted, which is basically a curve showing the optimum combinations of throughput rate and WIP. Not one of the solutions on the Pareto front is better or worse than any other solution on the front, but they are better than any other solution in the solution space. The Pareto front is shown in Figure 5.2 and the corresponding configurations that generated the data points for the front are shown in Table 5.7.

The Pareto front has many data points from CC3 using inspection method 2 while running in batch mode. As expected, inspection method 2 should be dominating inspection method 1 because the assembly process essentially is shortened by one step when using inspection method 2. However, inspection method 2 is

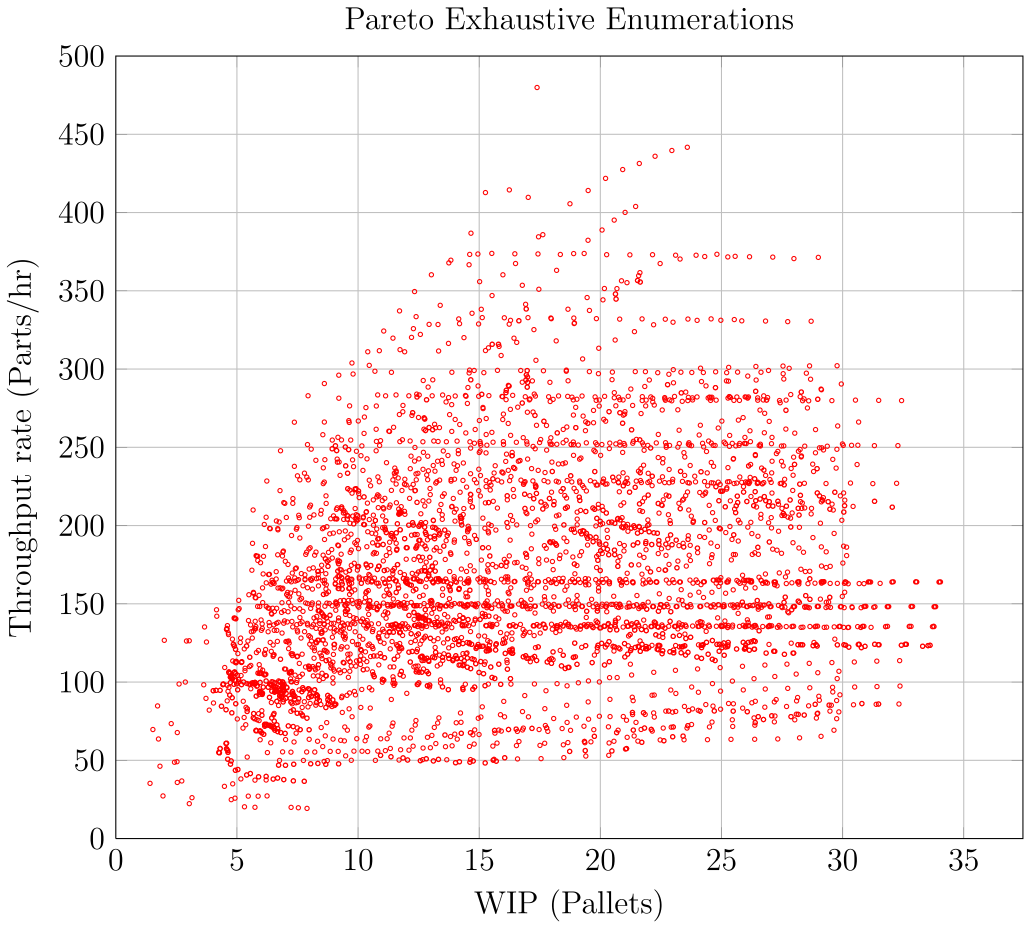


Figure 5.1: Pareto Exhaustive objective space.

more expensive than inspection method 1 because more cameras are required for machine vision.

Table 5.7: Pareto Front data points.

Conveyor Config	Pallets	Inspection Method	Batch/ Mix	Capacity	WIP	TPR
3	36	2	Batch	MaxCap	17.39	479.89
3	35	2	Batch	MaxCap	16.24	414.46
3	33	2	Batch	MaxCap	15.26	412.76

Continued on next page

Table 5.7 – continued from previous page

Conveyor Config.	Pallets	Inspection Method	Batch/ Mix	Capacity	WIP	TPR
3	31	2	Batch	MaxCap	14.66	386.80
2	24	2	Batch	MaxCap	14.61	373.25
3	30	2	Batch	MaxCap	13.83	369.55
2	23	2	Batch	MaxCap	13.75	367.85
2	22	2	Batch	MaxCap	13.03	360.23
2	21	2	Batch	MaxCap	12.33	349.51
2	20	2	Batch	MaxCap	11.71	337.15
2	19	2	Batch	3PF,3IR,2W	11.06	324.29
3	22	2	Batch	MaxCap	10.88	311.72
2	18	2	Batch	3PF,3IR,2W	10.40	311.04
3	17	2	Batch	3PF,3IR,2W	9.75	303.87
3	16	2	Batch	3PF,3IR,2W	9.20	296.08
3	15	2	Batch	3PF,3IR,2W	8.61	290.74
3	14	2	Batch	3PF,3IR,2W	7.94	282.94
3	13	2	Batch	3PF,3IR,2W	7.37	266.05
3	12	2	Batch	3PF,3IR,2W	6.80	247.78
3	11	2	Batch	3PF,3IR,2W	6.23	228.45
3	10	2	Batch	3PF,3IR,2W	5.67	209.95
3	10	2	Batch	MaxCap	5.64	188.64
3	10	2	Batch	MinCap	5.49	155.79
1	6	2	Batch	MaxCap	5.07	152.22
1	5	2	Batch	MaxCap	4.16	146.20
1	5	2	Batch	MaxCap	3.66	135.22
1	5	1	Batch	MaxCap	2.00	126.68
1	5	1	Batch	MaxCap	1.72	84.76
1	5	1	Batch	MaxCap	1.53	69.68
1	9	2	Batch	MaxCap	1.41	35.28

Table 5.7 also shows that batch mode is used in every solution and not mixed mode. One reason for this could be that the approximated batch delays are too short. Another reason is that during batch mode, all the Q-frames are assigned exactly the same number of pallets, regardless of their order size. The result is that the Q-frames with smaller orders will be completed quicker in batch mode than in mixed mode. Note from Table 4.4 on page 92 that the number of pallets

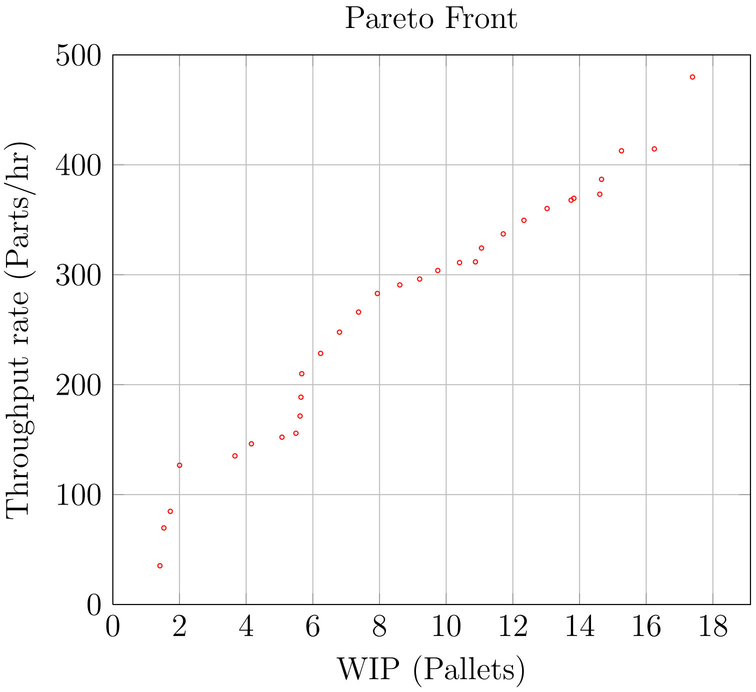


Figure 5.2: Pareto Front

assigned to Q4 and Q5 is always one due to their order size while in batch mode, they are assigned the same number of pallets than all the other frames. This definitely has a cost implication because more pallets and more fixtures are required to operate the system in batch mode.

In section 4.3 on page 84 it was explained that pallets are assigned two welding times. *Welding time 1* is when the welder has the correct tool to weld a particular frame that has arrived at the welder. If the welder has to change its tool, *welding time 2* is used. *Welding time 2* is assigned a triangular distribution of Triangular(13.5, 14, 14.2) seconds while *welding time 1* is assigned a triangular distribution of Triangular(8.5, 9, 9.2) seconds. Hence *welding time 2* takes approximately 56% longer than *welding time 1* due to the required tool change.

The two welding times will affect the system the most when it is running in mixed mode because then there are five different types of pallets on the system which results in regular tool changes. An experiment was done to determine what the effect on the TPR of the system is when no tool changes are required. The experiment was also done to determine if the regular tool changes are the reason that the *mixed* operating systems are outperformed by the *batched* operating systems.

Table 5.8 and Table 5.9 show the results of the experiment. *TPR Batch* refers to the throughput rate of CC2, for the specified capacity, when the system is running in batch mode. *TPR 1* is the throughput rate of CC2 when tool changes are used and *TPR 2* are when tool changes are not used. *% difference* is the percentage difference between TPR 1 and TPR 2.

For a higher number of pallets in the system, the percentage difference between TPR 1 and TPR 2 is higher because more tool changes are required when there are more pallets on the system. Looking at TPR batch and TPR 2, it is clear that the mixed systems out perform the batched systems when there are many pallets on the system. As the number of pallets decrease and the effect of tool changes are also less, the batch systems out perform the mixed systems once again.

The percentage difference in Table 5.8 is lower than that of Table 5.9. This is due to the capacity of the system. For maximum capacity when three welders are active, their utilization is on average 75.63% while if only one welder is active, its utilization is 96.44%. If the welders are used less, they will undergo fewer tool changes. Tool changes can also happen in parallel because three pallets can be welded at any time. With only one welder having to weld every pallet, the effect of tool changes are greater. Hence the effect of tool changes are less when the system is running at full capacity.

Tool changes have a more significant effect on the mixed mode systems, but the main reason that they are out performed by batch mode, is that there are more pallets of a particular type of Q-frame on the system at a time. This increases individual throughput rates of Q-frames but it is also more expensive because more pallets are required.

The purpose of the Pareto exhaustive experiment is to test all the possible scenarios so that if the decision maker is not satisfied with the Pareto front of Figure 5.2, the criteria for choosing the Pareto front can be changed. For example, the decision maker is not willing to pay extra for running the system with inspection method 2 and batch mode. These solutions can then be removed from the solution space, and a new Pareto front can easily be constructed because

Table 5.8: Throughput rate differences for maximum capacity.

Pallets	**TPR Batch**	**TPR 1**	**TPR 2**	**% difference**
35	227.45	229.86	265.16	15.36
31	216.26	188.07	216.11	14.91
23	182.25	139.13	154.99	11.4
16	136.71	73.03	80.10	9.687
12	104.09	69.01	74.95	8.615

Table 5.9: Throughput rate differences for 3PF, 3IR, 1W.

Pallets	**TPR Batch**	**TPR 1**	**TPR 2**	**% difference**
35	134.94	123.75	153.06	23.68
31	135.10	115.99	140.90	21.47
23	135.72	106.14	126.95	19.61
16	134.06	71.53	84.91	14.69
12	103.88	68.87	81.38	13.94

all the possibilities have already been calculated. Hence the Pareto exhaustive experiment is there to inform the decision maker of what is possible rather than prescribing what to do. In essence, the Pareto front can be tailored to fit the needs of the decision maker.

In addition to the Pareto front constructed from the solution space of Figure 5.1, Pareto fronts were also constructed for each conveyor configuration variation. The reason again is to inform the decision maker of what is possible. The Pareto front is constructed using a ranking algorithm and two examples are shown in Figure 5.3 and Figure 5.4.

Inspection method 2 and batch mode were used to generate the data for Figure 5.3 while inspection method 2 and mixed mode was used to create Figure 5.4. The figures first show the data obtained through enumerations and then the Pareto front extracted from the data. The Pareto fronts and enumerations of the other configurations can be viewed on the CD.

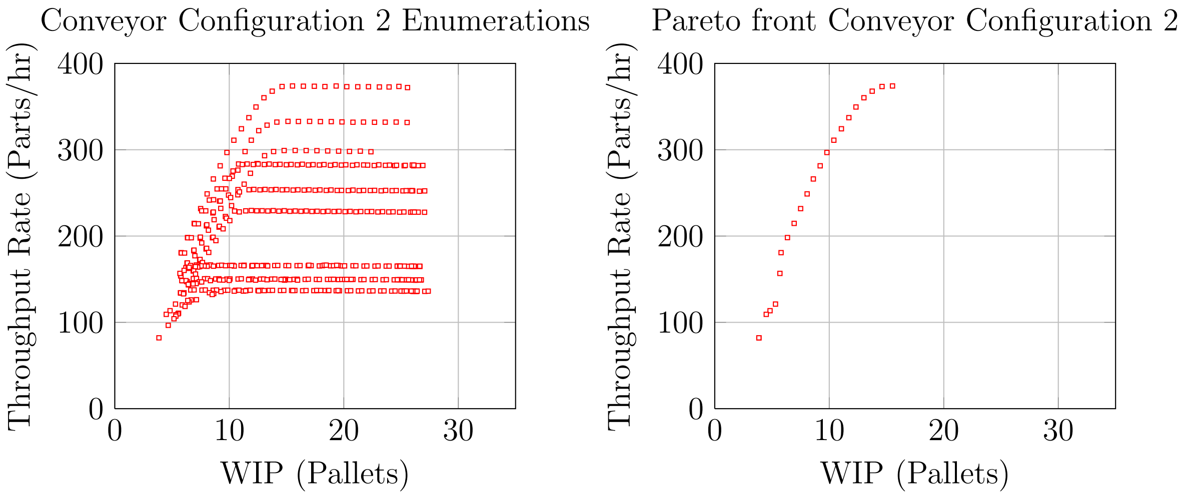


Figure 5.3: Enumerations and Pareto front for Conveyor Configuration 2.

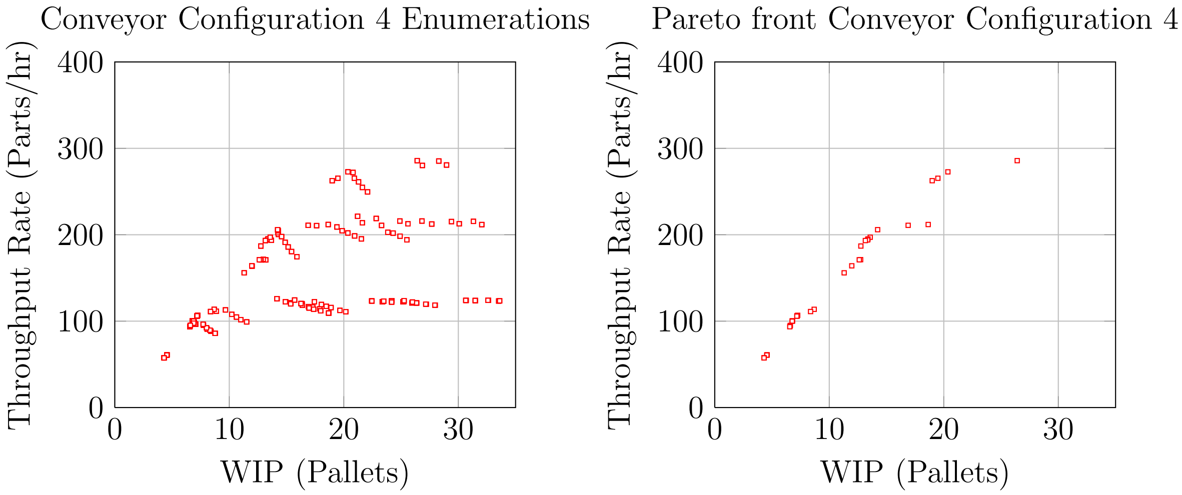


Figure 5.4: Enumerations and Pareto front for Conveyor Configuration 4.

5.3 Results of Conveyor Speed Adjustment Experiment

In section 4.6 it was stated that the reason the conveyor speed experiment was done was to investigate the phenomenon of the systems delivering the exact same throughput rate at different conveyor speeds, when running in mixed mode. This discovery was made during the Pareto exhaustive experiment and it can be seen from Figure 5.1 that there are data points that overlap each other. To better understand why the phenomenon occurred, the conveyor speed experiment was conducted to test different conveyor speeds with different numbers of pallets. The results obtained from the experiment can be seen in Figure 5.5 and Figure 5.6.

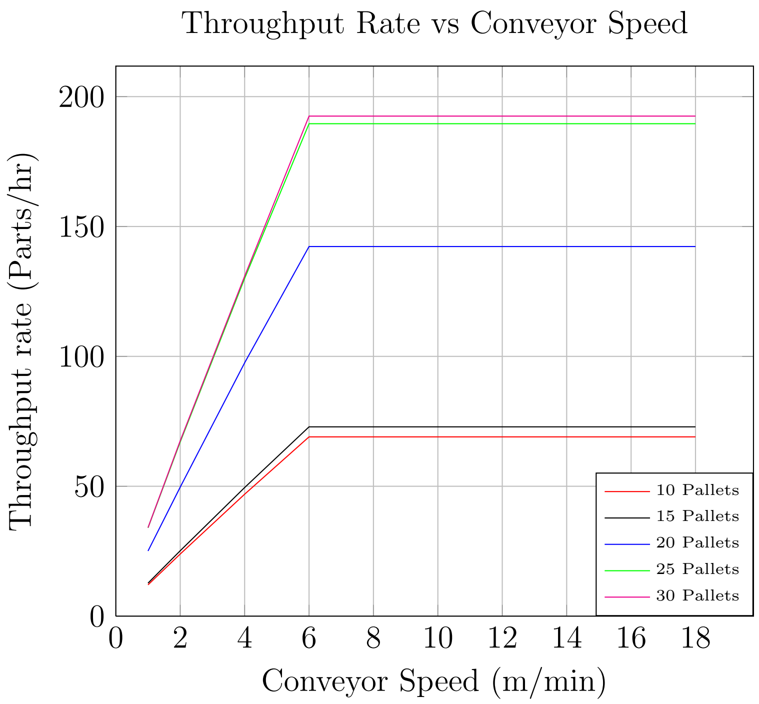


Figure 5.5: Conveyor speeds for CC2.

From both figures it is clear that conveyor speeds higher than 6 m/min has no effect on the throughput rate of the system when the system is operating in mixed mode. Tables 5.10, 5.11, 5.12, and 5.13 show the throughput rates for the individual Q-frames, and that of the system, when the system is working with a total of 15, 20, 25, and 30 pallets respectively. Recall from Table 4.6 on page 93 that the pallets are assigned proportionally to each type of Q-frame according to their order sizes. Due to the low order sizes of Q4 and Q5, they are never assigned more than one pallet, while the other Q-frames are assigned more pallets as the total number of pallets in the system increases.

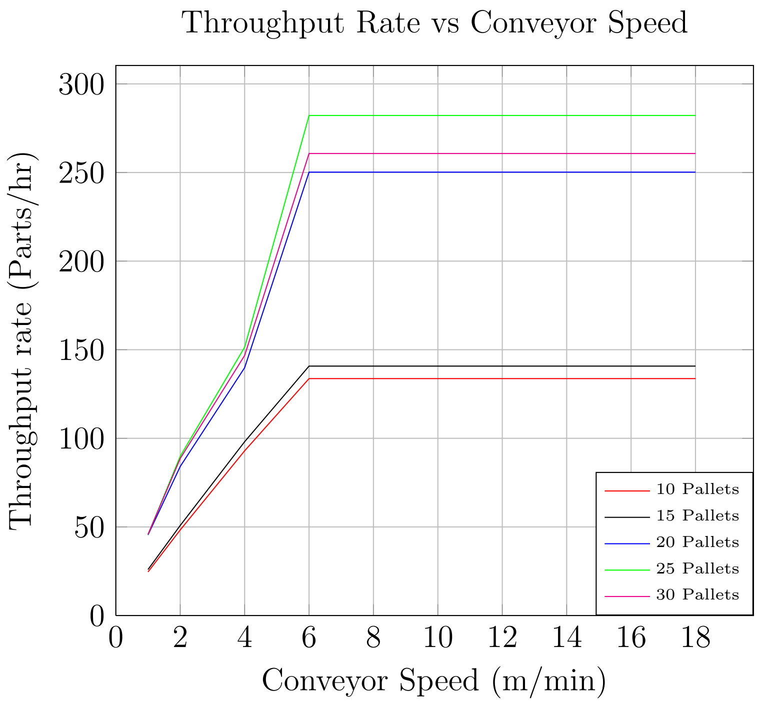


Figure 5.6: Conveyor speeds for CC3.

At conveyor speeds lower than 6 m/min, the system is constrained by the conveyor speed because the pallets are not being transported quickly enough to the different stations. The result is work stations standing idle and waiting for pallets to arrive. Note the low throughput rates in Table 5.10 at conveyor speeds of 1 and 2 m/min.

For conveyor speeds higher than 6 m/min, it is not the conveyor speed that is the constraint but rather the number of pallets assigned to Q4 and Q5. Remember that the throughput rate of the system takes the throughput rate of all five different Q-frames into account. Since the system operates on a FIFO production sequence, the single pallet of Q4 and Q5 each has to compete with many pallets of the other Q-frames. Hence with the current machine processing times, a conveyor speed greater than 6 m/min does not help the pallets of Q4 and Q5 to be served quicker. This is evident in Tables 5.10 to 5.13 since the throughput rates of Q4 and Q5 are almost the same regardless of the total number of pallets assigned to the system. The throughput rates of Q1, Q2, and Q3 do however increase because the number of pallets assigned to them increases. Hence the single pallets of Q4 and Q5 causes the system to reach a ceiling value for its throughput rate at conveyor speeds of 6 m/min and higher.

Note in Figure 5.5 that there is a large difference between the system throughput rates when the system is working with 15 and 20 pallets respectively. There is also a large difference when the system is working with 20 and 25 pallets respectively. The reason for this is the increase in number of pallets assigned to Q1, Q2 and especially Q3. Q3 is assigned 1 pallet when the system is working with 10 and 15 pallets in total. The throughput rate of the system between 10 and 15 pallets only differ slightly as shown in Figure 5.5. When the system is working with 20 pallets, Q3 is assigned 2 pallets instead of only 1. Note the difference in Q3 throughput rate in Table 5.10 and Table 5.11. It has doubled due to the extra pallet assigned to Q3. The throughput rates of Q4 and Q5 remain low because they are only assigned 1 pallet. Q1 and Q2 throughput rates also increase but not as drastically as that of Q3. This supports the argument that the system is constrained by the number of pallets assigned to the different Q-frames and not the conveyor speed for a conveyor speed higher than 6 m/min.

The same explanation holds for the large difference in throughput rates between 20 and 25 pallets. Q3 pallets increase from 2 to 3 which causes a relatively large increase in its throughput rate. The system throughput rate of the system differ slightly between 25 and 30 pallets because the number of pallets assigned to Q3 remain the same and because the system is close to its maximum capacity for the number of pallets that can be on the system at the same time.

In batch mode there is only one type of Q-frame on the system at a time. Hence there is not a constraint on the number of pallets assigned to a specific Q-frame. All the Q-frames have the same number of pallets on the system when that batch is produced. Hence as the conveyor speed increases, the throughput rate will also increase.

The conveyor speed has a cost implication because for higher conveyor speeds, larger motors that drive the conveyor need to be installed. The decision maker has to take this into account when choosing which type of operating mode (mixed vs batch) and which conveyor speed to run the system at.

Table 5.10: Throughput rates for 15 Pallets.

	Throughput Rates					
Conveyor Speed	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**System**
1	12.15	6.07	1.53	1.52	1.51	12.71
2	24.08	12.03	3.02	2.99	3.01	25.18
4	47.18	23.61	5.94	5.89	5.88	49.51
6	69.39	34.63	8.75	8.62	8.62	72.88
9	69.39	34.63	8.75	8.62	8.62	72.88
18	69.39	34.63	8.75	8.62	8.62	72.88

Table 5.11: Throughput rates for 20 Pallets.

	Throughput Rates					
Conveyor Speed	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**System**
1	16.44	7.50	3.01	1.49	1.48	25.09
2	32.58	14.85	5.96	2.96	2.95	49.68
4	63.43	28.94	11.70	5.78	5.69	97.52
6	92.73	42.28	17.07	8.40	8.33	142.29
9	92.73	42.28	17.07	8.40	8.33	142.29
18	92.73	42.28	17.07	8.40	8.33	142.29

Table 5.12: Throughput rates for 25 Pallets.

	Throughput Rates					
Conveyor Speed	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**System**
1	20.05	8.62	4.31	1.42	1.42	34.01
2	39.43	17.03	8.51	2.81	2.81	67.07
4	76.35	33.05	16.47	5.50	5.39	130.03
6	111.28	48.18	24.06	8.01	7.93	189.57
9	111.28	48.18	24.06	8.01	7.93	189.57
18	111.28	48.18	24.06	8.01	7.93	189.57

5.4 Results of Ramp-up Time Experiment

All the experiments conducted before the ramp-up time experiment tested the effects of different capacity combinations. Although different capacity combina-

Table 5.13: Throughput rates for 30 Pallets.

	Throughput Rates					
Conveyor Speed	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**System**
1	22.68	10.61	4.09	1.32	1.30	34.08
2	44.57	20.86	8.10	2.60	2.59	67.52
4	86.62	40.55	15.70	5.09	5.01	130.87
6	126.24	59.25	23.10	7.42	7.35	192.50
9	126.24	59.25	23.10	7.42	7.35	192.50
18	126.24	59.25	23.10	7.42	7.35	192.50

tions were tested, the system started in an initial capacity state that remained unchanged during a run. The fact that the system can be tested at different capacity combinations, does not make it a reconfigurable manufacturing system. The system must be able to change its capacity quickly, without causing a major disruption, in order to be classified as a reconfigurable manufacturing system. Hence in the ramp-up time experiment, the aim was to test how the system reacts to changing its capacity.

The production scenario that was outlined in section 4.7 on page 93 was replicated 100 times, since there is a large variation in the reconfiguration delays of Table 4.9 on page 95. The first experiment response that was measured is the sum of the ramp-up times during a single run. Recall from section 4.2 on page 82 that there are three types of ramp-up times that are added together to calculate the sum of the ramp-up times.

An experiment of 100 replications was done for each of the conveyor configuration variations. Hence for each variation there are 100 observations for the sum of the ramp-up times. A histogram can be drawn from the 100 observations from which the cumulative distribution for each configuration's sum of ramp-up times can be calculated. Figure 5.7 and Figure 5.8 show the cumulative distributions for the sum of the ramp-up times for when the configurations are running in mixed and batched mode respectively. In the legend, I1 and I2 refer to inspection method 1 and 2 respectively.

The cumulative distribution is a useful tool which the decision maker can use during production planning. Using Figure 5.7 as an example, it can be seen that

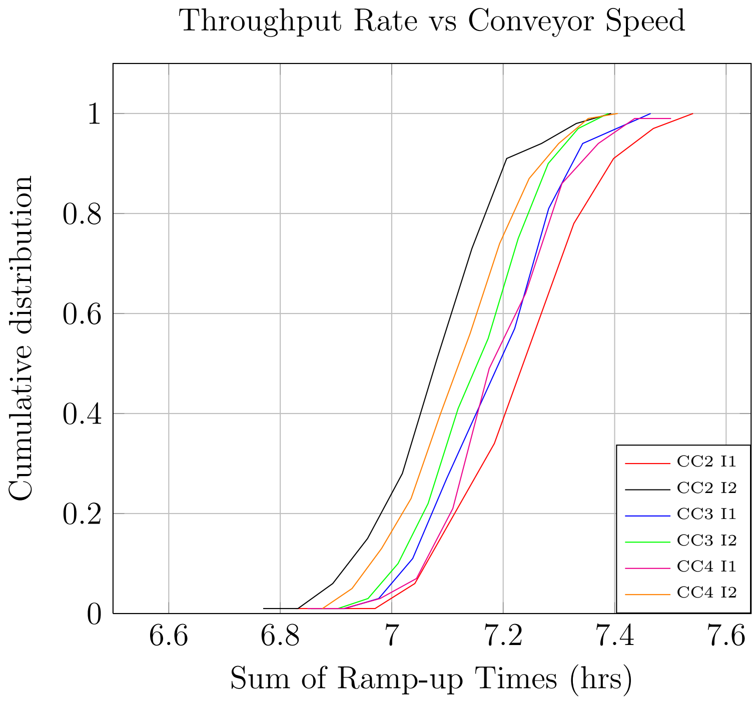


Figure 5.7: Cumulative ramp-up time distributions for mixed conveyor configurations.

the sum of the ramp-up times of CC2 I2 will for 40% of the time not exceed 7.2 hours. Using these probabilities and factoring in some unexpected delays, the decision maker will be able to estimate the completion time of the given production scenario.

Obviously, the cumulative distributions shown in Figure 5.7 and Figure 5.8 are only relevant and valid for the given production scenario and reconfiguration delays. If the production scenario or the reconfiguration delays are altered, new experiments must be conducted. For details on each of the individual configurations, refer to the CD.

The throughput rate of the system and the WIP that were calculated in the previous experiments, did give some good insight into the characteristics of the configurations. However, the systems were not operating according to a production scenario. For this reason, the throughput rate and WIP for all the configuration variations are shown in Figure 5.9 in order to compare the

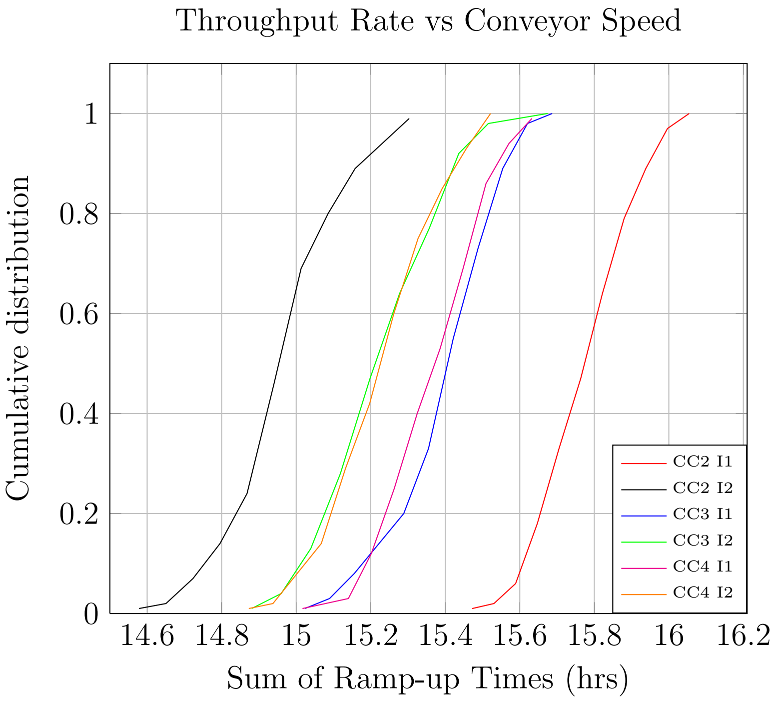


Figure 5.8: Cumulative ramp-up time distributions for batch conveyor configurations.

configurations to each other.

From Figure 5.9 it is clear that the best combination of throughput rate and WIP is conveyor configuration 3, using inspection method 2 and running in batch mode. Remember that this conveyor configuration configuration also dominated the Pareto front that was developed in section 5.2.

For each of the conveyor configurations, the throughput rate and WIP were also plotted. The reason for this is that if the decision maker decides on a specific configuration, it is possible to choose the correct operating logic for that particular system. For example, in Figure 5.10 the decision maker can choose between the different operating variations of CC2. If the decision maker is willing to pay extra for inspection method 2, running the system in mixed mode is recommended because it is cheaper than running the system in batched mode and the WIP is lower. If the decision maker chooses inspection method 1, it is obvious that mixed mode should be the operating logic of choice. There is little difference between

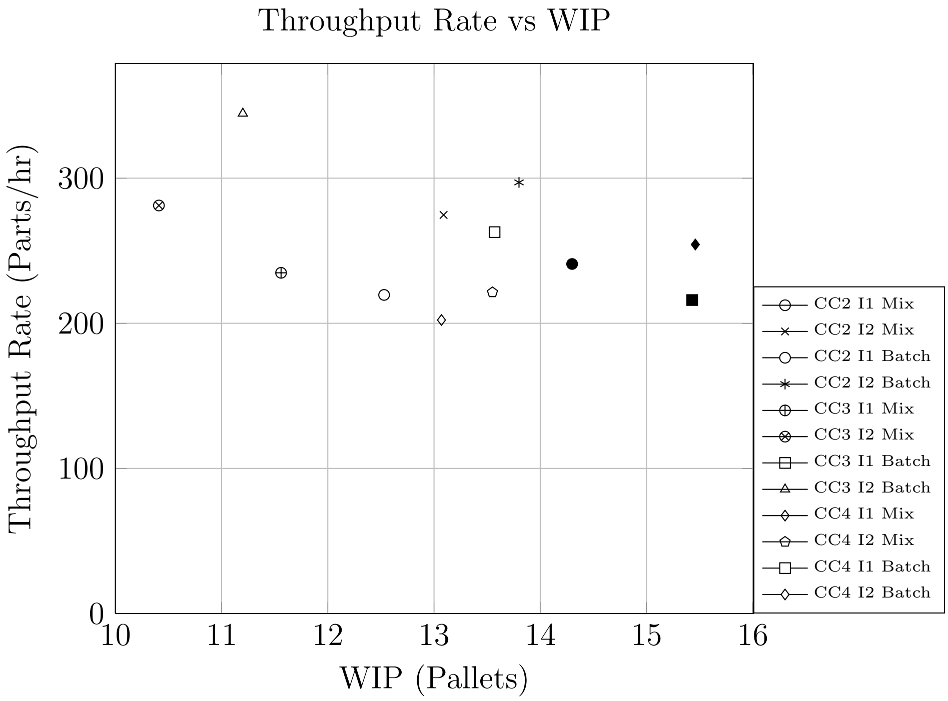


Figure 5.9: Summary of throughput rate vs WIP for ramp-up time experiment.

the throughput rate for mixed and batch, but a large difference in the WIP of the system.

5.5 Chapter Summary

In this chapter the results obtained from the different experiments were presented. From the discussion of these results, insight was gained about the characteristics of the developed configurations. In the next chapter conclusions and recommendation will be made regarding the research that was performed during this project.

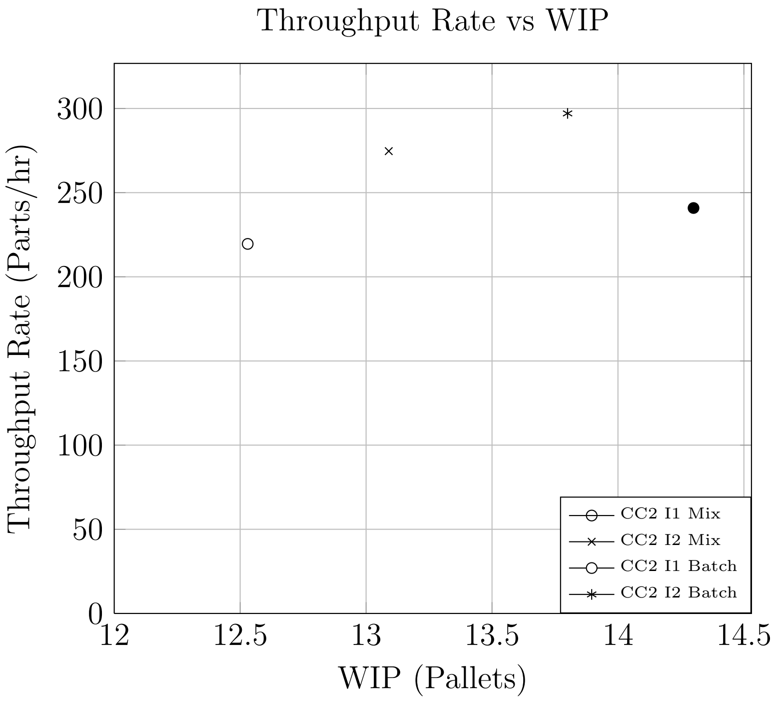


Figure 5.10: Throughput rate vs WIP for CC2.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

In the previous chapter, the results obtained from the four different experiments were discussed and the knowledge gained makes it possible to draw some conclusions regarding the different conveyor configurations.

6.1 Conclusions regarding the initial conveyor configuration

First of all it is important to realize that the initial conveyor configuration cannot be compared to the proposed configurations that were used in all the experiments. The reason being that CC1 is not a reconfigurable manufacturing system because it cannot add or remove capacity. CC1 can be viewed more as a flexible manufacturing system because the machinery are flexible enough to be able to assemble a part family.

CC1 can only be compared to the other configurations when they are operating at minimum capacity, which is CC1's maximum capacity. This comparison is still flawed, because the aim of this research is to study reconfigurable manufacturing systems. It is therefore concluded that CC1 was used as a starting point to learn the simulation software and to serve as a basis for developing the other conveyor configurations.

6.2 Conclusions regarding capacity

The first conclusion that can be drawn from the validation experiment is that the three proposed conveyor configurations are indeed different. They differ in layout and more importantly differ statistically based on throughput rate. The configurations were also validated to ensure that the systems react as expected to changes in capacity.

From the validation experiment it can be concluded that it is good practice to ensure that there are more part feeders and inspection & removal stations than welders in the system. The reason being that welders are expensive, and therefore it is desirable to utilize the welders as much as possible. The welders are also the bottleneck which determines the cycle time of the system.

The conveyor configurations are currently designed to be able to house three system modules of each type of machine. If it is decided that there must be more part feeders and inspection & removal stations than welders, the layout of the conveyors can be altered to make the system even more efficient. It could then be possible to deliver the same level of throughput rate, with a lower WIP, resulting in less pallets that are required.

6.3 Conclusions regarding operating logic

Initially CC2, CC3, and CC4 had eight variations based on operating logic. It was then determined that the sequence in which parts are placed is insignificant, which resulted in the configurations having only four variations.

The costs associated with each variation were not exactly calculated but reasonable assumptions can be made to compare each variation to each other. The choice of inspection method requires either more or less cameras for machine vision. Inspection method 1 only needs cameras at the inspection & removal stations because all the inspections (before and after welding) are done at these stations. Inspection method 2 on the other hand requires additional cameras at the part feeders because the inspection step before welding is completed at the part feeding stations. It can therefore be concluded that although inspection method 2 has one less assembly step and delivers a higher throughput rate, it is more expensive than inspection method 1.

Running the system in batch mode generally delivers a higher throughput rate compared to running the system in mixed mode. As previously stated in section 5.2, more pallets and more fixtures are required for running the system in batch mode. The result is that batch mode will on average be more expensive than mixed mode.

6.4 Conclusions regarding experiments

The Pareto exhaustive experiment was performed to inform the decision maker of what the configurations are capable of achieving. The proposed Pareto front consisted mainly of the more expensive possibilities but it was also the best possibilities. The decision maker must decide what is *best* for his application and what solution fits the budget. It can be concluded that the decision maker can tailor a solution that best fits his needs using the results of the Pareto exhaustive experiment.

The conveyor speed experiment made it clear that when the configurations are operating in mixed mode, it is unnecessary to increase the conveyor speed higher than 6 m/min. This is only valid for the way the pallets were assigned to the different Q-frames. If more pallets are assigned to the Q-frames with lower orders, the conveyor speed will have a greater effect on the overall throughput rate. The conveyor speed also has a cost implication that the decision maker should consider when choosing a conveyor configuration.

Finally the ramp-up time experiment gave insight into what the expected delays of a system can be for the given production scenario. The systems running in batch mode had higher ramp-up times than the systems running in mixed mode. However, the systems should be compared to each other based on the throughput rate and WIP as shown in Figure 5.9. From the figure it can be concluded that CC3 using inspection method 2 and running in batch mode, is the overall best solution. However, the decision maker must still decide whether he/she is willing to pay the required costs.

6.5 Project Aims Reached

Recall from section 2.10 on page 52 that the aims of this study were outlined. A better understanding of the characteristics of the developed configurations were indeed established. The experiments that were conducted gave insight into how the systems react to capacity changes, conveyor speed adjustments, and different production scenarios to name a few. It is now known which variables of the system control, and have an effect on the output.

Multi-objective optimization was applied through the use of the Pareto exhaustive search experiment. Two objectives were also used during the ramp-up time experiment. With multi-objective optimization, a more informed decision can be made when choosing between the different configurations.

Simulation models were created to act as a Decision Support System (DSS), that can be used by the machine and control system designers to test their designs once completed. The simulation models focus on the operation of the system and not the detail of each individual machine. Hence the designers can study the effect a change in their design has on the entire system. For example, a change in the machine processing time can easily be tested with the current simulation models.

Finally as a platform for future researchers to work from, guidelines for the simulation models are presented in Appendix A. The purpose of the guidelines are to make it easier for a new user to understand the models, and to make it possible for them to change the experimental controls with ease. It is however advisable that the user read introductory notes for Simio simulation software.

6.6 Recommendations

When deciding on a configuration, the first step is to decide what the maximum capacity of the system will be and which capacity combinations will be used during production. The chosen configuration will be replicated many times on the factory floor and these subsystems will take machine modules out of a pool of resources.

Next the conveyor layout must be chosen out of the three proposed designs. The layouts can be altered slightly if the chosen maximum capacity is less than the current designs. A slight alteration can include changes like a shorter conveyor belt. For the chosen configuration, the data generated in the experiments can be used to determine which operating logic will be used during production.

Future work that must be done is to compare the simulation model of the initial conveyor configuration to the real world working system that is under construction at the Department of Mechanical and Mechatronic engineering. This is a good validation exercise and it can be used to scrutinize the proposed designs.

Future research on the topic of reconfigurable manufacturing systems can be to do a simulation study where the conveyor systems are replaced with Automated Guided Vehicles (AGV). The same product can still be assembled with the same system modules except that the pallets are transported by AGVs and not a conveyor belt.

REFERENCES

ABDI, M.R. & LABIB, A.W. (2003). A design strategy for reconfigurable manufacturing systems (RMSs) using analytical hierarchical process (AHP): a case study. *International Journal of Production Research*, **41**, 2273 – 2299. 4, 5, 7

ABDI, M.R. & LABIB, A.W. (2004). Grouping and selecting products: the design key of reconfigurable manufacturing systems (RMSs). *International Journal of Production Research*, **42**, 521 – 546. 12, 13

ALTIOK, T. & MELAMED, B. (2007). Output analysis. In *Simulation Modeling and Analysis with ARENA*, 165 – 194, Academic Press, Burlington. 39, 40, 41, 43

BANKS, J. (1998). *Handbook of Simulation*. John Wiley & Sons Inc. 27, 28, 35, 76

BEKKER, J. & ALDRICH, C. (2011). The cross-entropy method in multi-objective optimisation: An assessment. *European Journal of Operational Research*, **211**, 112 – 121. 48

BEKKER, J.F. (2009). *Discrete-event Simulation*. Stellenbosch University. 33, 42, 75

BI, Z.M., LANG, S.Y.T., SHEN, W. & WANG, L. (2008). Reconfigurable manufacturing systems: the state of the art. *International Journal of Production Research*, **46**, 967 – 992. 8, 9, 10

COELLO COELLO, C. (2006). Evolutionary multi-objective optimization: a historical view of the field. *Computational Intelligence Magazine, IEEE*, **1**, 28 – 36. 48

GROOVER, M.P. (2008). *Automation, Production Systems, and Computer-Integrated Manufacturing*. Prentice Hall International, 3rd edn. 1, 4

JOHN W. TUKEY, W.A.L., ROBERT MCGILL (1978). Variations of box plots. *The American Statistician*, **32**, 12–16. 46, 47

KATZ, R. (2007). Design principles of reconfigurable machines. *International Journal of Advanced Manufacturing Technology*, **34**, 430 – 439. 14, 15, 16

KOREN, Y., HEISEL, U., JOVANE, F., MORIWAKI, T., PRITSCHOW, G., ULSOY, G. & BRUSSEL, H.V. (1999). Reconfigurable manufacturing systems. *CIRP Annals - Manufacturing Technology*, **48**, 527 – 540. 4, 5, 6, 7, 9, 10, 11

LAW, A.M. (2007). *Simulation Modeling and Analysis*. McGraw-Hill Inc., 4th edn. 37, 38, 41, 44, 45, 75, 76, 77, 98

MEHRABI, M., ULSOY, A. & KOREN, Y. (2000). Reconfigurable manufacturing systems: Key to future manufacturing. *Journal of intelligent Manufacturing*, **11**, 403–419. 5, 6, 7, 10, 21, 24

NELSON, B.L. (2008). The more plot: Displaying measures of risk & error from simulation output. In *Winter Simulation Conference*, 413–416. 46, 47

SEQUEIRA, M.A. (2008). *Conceptual Design of a Fixture-Based Reconfigurable Spot Welding System*. Master's thesis, Stellenbosch University. 55, 56, 59

SPICER, P. & CARLO, H.J. (2007). Integrating reconfiguration cost into the design of multi-period scalable reconfigurable manufacturing systems. *Journal of Manufacturing Science and Engineering*, **129**, 202–210. 16, 23, 24, 25, 26

TAM, K.Y. (1990). An operation sequence based similarity coefficient for part families formations. *Journal of Manufacturing Systems*, **9**, 55 – 68. 13

TANG, Y. & QIU, R.G. (2004). Integrated design approach for virtual production line-based reconfigurable manufacturing systems. *International Journal of Production Research*, **42**, 3803 – 3822. 2, 18, 19, 21, 22, 23

TELEB, R. & AZADIVAR, F. (1994). A methodology for solving multi-objective simulation-optimization problems. *European Journal of Operational Research*, **72**, 135 – 145. 48

W. DAVID KELTON, D.T.S.A.V., JEFFREY S. SMITH (2010). *Simio & Simulation: Modeling, Analysis, Applications*. McGraw-Hill. 46, 47

XIAOBO, J.W.Z.L., Z. (2000). A stochastic model of a reconfigurable manufacturing system. part 1: A framework. *International Journal of Production Research*, **38**, 2273–2285. 14

APPENDIX A

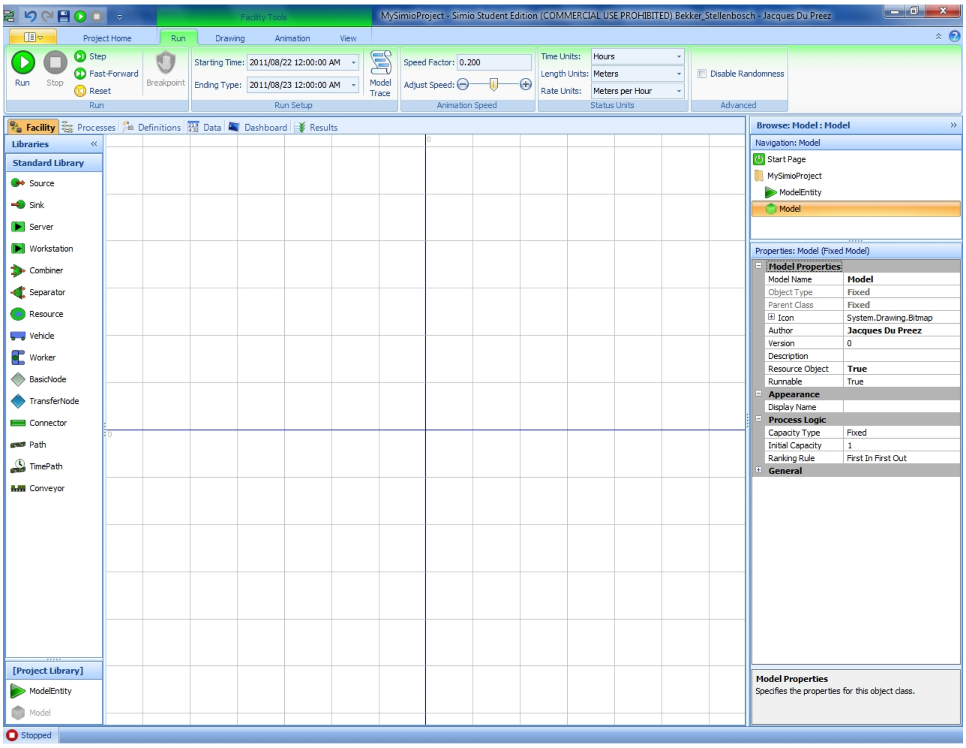
SIMIO GUIDELINES

The purpose of this guide is not to teach Simio to the reader, but rather give guidelines on how to use the models that were developed during this research project. It is therefore important to study the introduction to Simio guide which is provided with the software.

A.1 User Interface

The initial Simio project window is shown in Figure A.1. The key areas in Figure A.1 include the ribbons across the top (currently showing the Run ribbon), the tabbed panel views with the Facility highlighted just below the ribbons, the libraries on the left, the browse panel on the right, and the Facility window in the center.

The ribbons are tabbed panels that are used to access many functions like building, running and animating the models. The browse panel on the right is used for project navigation and model property editing. The upper navigation window is used to switch between the Start Page, Project view, and the associated models and experiments. All the sub-models that are developed during a project can be viewed in the navigation window. Clicking on a model will make it the active model in the Facility window. Below the navigation window is the property window that is used to edit the properties of objects. Each type of object has



MySimioProject - Simio Student Edition (COMMERCIAL USE PROHIBITED) Bekker_Stellenbosch - Jacques Du Preez

Starting Time: 2011/08/22 12:00:00 AM
Ending Type: 2011/08/23 12:00:00 AM
Speed Factor: 0.200
Adjust Speed:
Time Units: Hours
Length Units: Meters
Rate Units: Meters per Hour
☐ Disable Randomness

Properties: Model (Fixed Model)

Model Properties	
Model Name	**Model**
Object Type	Fixed
Parent Class	Fixed
Icon	System.Drawing.Bitmap
Author	**Jacques Du Preez**
Version	0
Description	
Resource Object	**True**
Runnable	True
Appearance	
Display Name	
Process Logic	
Capacity Type	Fixed
Initial Capacity	1
Ranking Rule	First In First Out
General	

Model Properties
Specifies the properties for this object class.

Figure A.1: Simio project window.

their unique properties and selecting an object in the facility window will show its properties in the properties window.

Whenever the Facility window of a model is selected the Libraries panel on the left displays the libraries that are open and available for modeling within the facility. The libraries will include the Standard Library, the Project Library, and any additional projects that have been loaded as libraries from the Project Home ribbon. Sub-models that are developed independently from the main model can be placed into the main model using the Project library window.

The Facility window is the space where the object-based model is built and animated. Other windows that are used to give extra functionality to the model are the processes, definitions, and data windows. The dashboard window can be used to monitor the model while running and the results window shows the results obtained from a run.

With a basic knowledge of the user interface, the following section will focus on how to change the experiment controls that were used during this project.

A.2 Changing the experiment controls

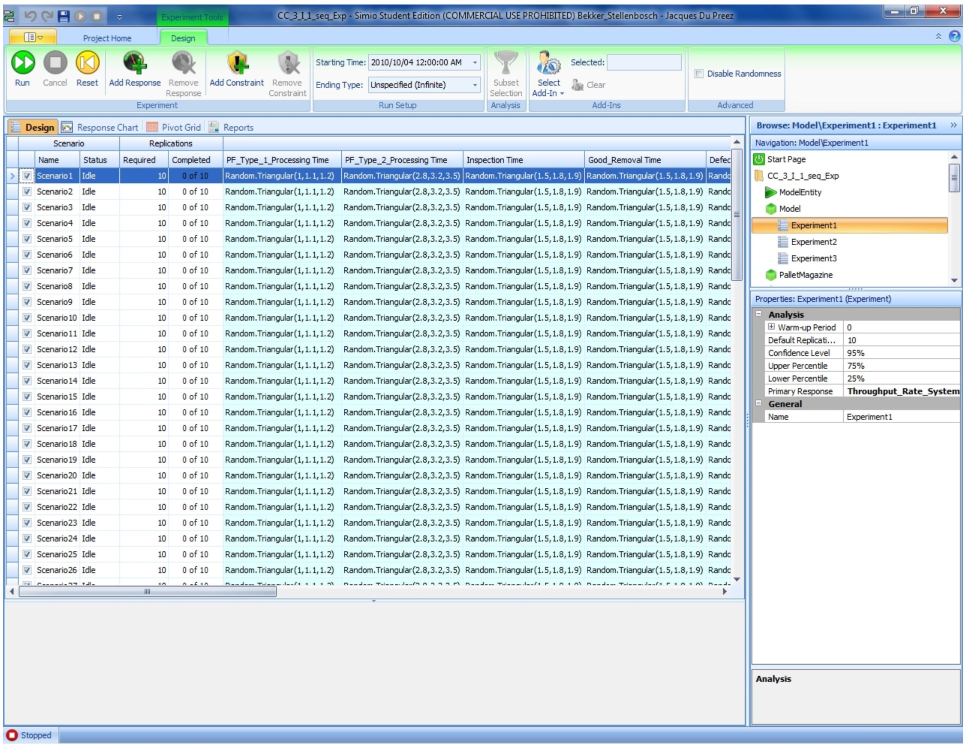
Recall from section 4.1 on page 79 that the controls for the experiments were the following:

- System Capacity
- Order
- Pallets
- Machine Processing times
- Conveyor Speed
- Batch Delays
- Reconfiguration Delays
- Maintenance

It is important to note that the models can be run in two different ways. The first of which is running the model in the Facility window. During this type of run, all the animations are visible and the speed of the run can be adjusted. This mode of running is used to ensure that the model is built correctly i.e. verifying the model.

Once the model is verified, experiments are conducted. The experiments can be viewed by clicking on an experiment in the navigation window. An example of the experiment user interface is shown in Figure A.2. In the experiment window, different scenarios can be developed by changing the above mentioned experiment controls. Each scenario can also be replicated many times which is not possible when running the model in the Facility window. A run is also completed much quicker using the experiments because no animations are required. The experiments deliver the output data that is used in the statistical analysis.

To view the output generated by an experiment, click on the pivot grid tab, as shown in Figure A.3. The pivot grid can be used to filter the output data to view only certain output statistics. The pivot grid can be exported to Excel



Starting Time: 2010/10/04 12:00:00 AM

Ending Type: Unspecified (Infinite)

	Scenario		Replications						
	Name	Status	Required	Completed	PF_Type_1_Processing Time	PF_Type_2_Processing Time	Inspection Time	Good_Removal Time	Defec
☑	Scenario1	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario2	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario3	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario4	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario5	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario6	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario7	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario8	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario9	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario10	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario11	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario12	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario13	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario14	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario15	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario16	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario17	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario18	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario19	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario20	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario21	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario22	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario23	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario24	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario25	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando
☑	Scenario26	Idle	10	0 of 10	Random.Triangular(1,1.1,1.2)	Random.Triangular(2.8,3.2,3.5)	Random.Triangular(1.5,1.8,1.9)	Random.Triangular(1.5,1.8,1.9)	Rando

Properties: Experiment1 (Experiment)

Analysis	
Warm-up Period	0
Default Replicati...	10
Confidence Level	95%
Upper Percentile	75%
Lower Percentile	25%
Primary Response	**Throughput_Rate_System**
General	
Name	Experiment1

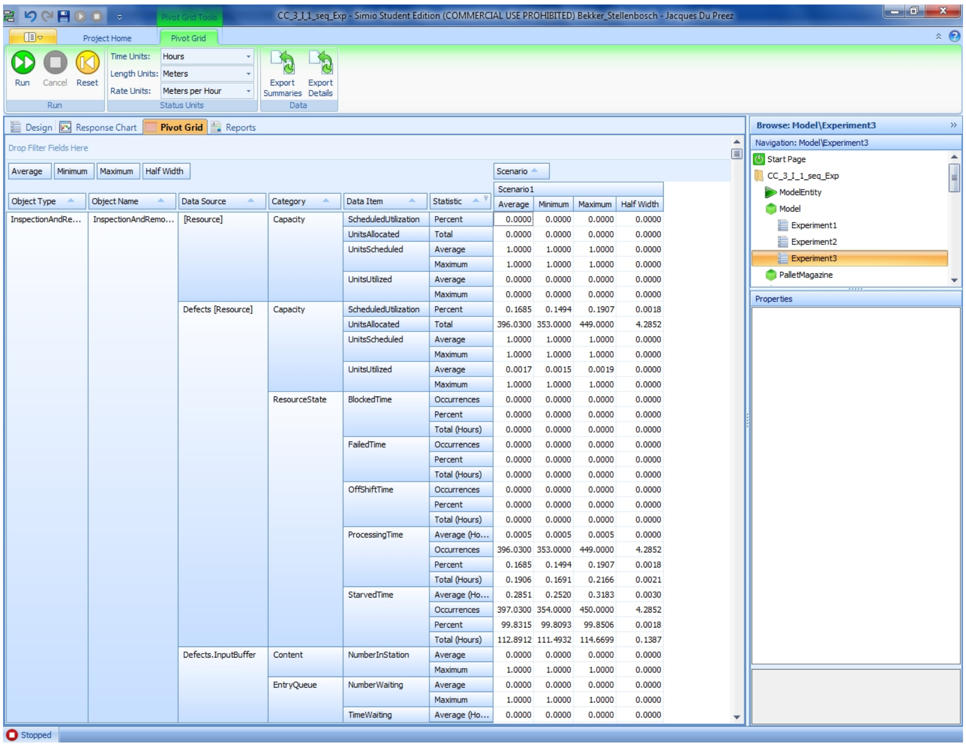
Figure A.2: Simio experiment window.

in two ways. In the ribbons tab there are two buttons, *Export Summaries* and *Export Details*. *Export Summaries* will export the average value for statistics across replications to a .csv file which can be used in Excel. *Export Details* will export all the values for statistics for each replication. It results in a much larger .csv file, but it is useful when doing the paired-t confidence interval analysis.

Not all the controls were changed during each experiment because the experiments were designed to each test specific characteristics of the configurations. Hence in the following sections, each experiment will be discussed in conjunction with the controls that were used for that particular experiment.

A.2.1 Controls for the Validation Experiment

The one set of controls that were never changed during all the experiments were the machine processing times. Clicking on the definitions tab and then on the properties panel on the left will display the different machine processing times



Time Units: Hours
Length Units: Meters
Rate Units: Meters per Hour

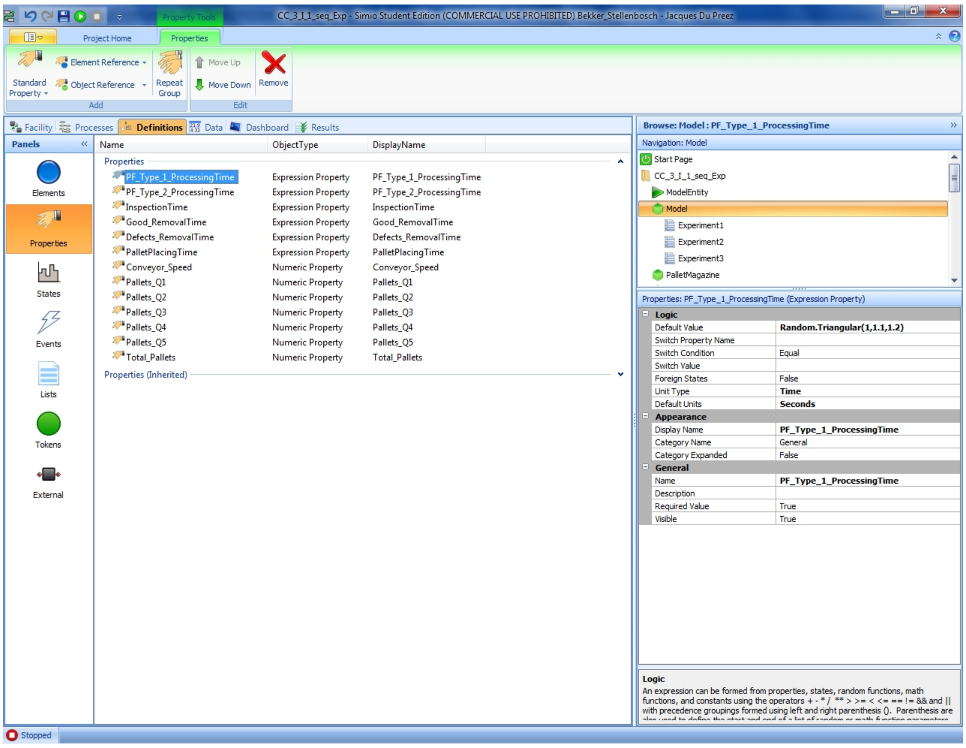
						Scenario1			
Object Type	Object Name	Data Source	Category	Data Item	Statistic	Average	Minimum	Maximum	Half Width
InspectionAndRe...	InspectionAndRemo...	[Resource]	Capacity	ScheduledUtilization	Percent	0.0000	0.0000	0.0000	0.0000
				UnitsAllocated	Total	0.0000	0.0000	0.0000	0.0000
				UnitsScheduled	Average	1.0000	1.0000	1.0000	0.0000
					Maximum	1.0000	1.0000	1.0000	0.0000
				UnitsUtilized	Average	0.0000	0.0000	0.0000	0.0000
					Maximum	0.0000	0.0000	0.0000	0.0000
		Defects [Resource]	Capacity	ScheduledUtilization	Percent	0.1685	0.1494	0.1907	0.0018
				UnitsAllocated	Total	396.0300	353.0000	449.0000	4.2852
				UnitsScheduled	Average	1.0000	1.0000	1.0000	0.0000
					Maximum	1.0000	1.0000	1.0000	0.0000
				UnitsUtilized	Average	0.0017	0.0015	0.0019	0.0000
					Maximum	1.0000	1.0000	1.0000	0.0000
			ResourceState	BlockedTime	Occurrences	0.0000	0.0000	0.0000	0.0000
					Percent	0.0000	0.0000	0.0000	0.0000
					Total (Hours)	0.0000	0.0000	0.0000	0.0000
				FailedTime	Occurrences	0.0000	0.0000	0.0000	0.0000
					Percent	0.0000	0.0000	0.0000	0.0000
					Total (Hours)	0.0000	0.0000	0.0000	0.0000
				OffShiftTime	Occurrences	0.0000	0.0000	0.0000	0.0000
					Percent	0.0000	0.0000	0.0000	0.0000
					Total (Hours)	0.0000	0.0000	0.0000	0.0000
				ProcessingTime	Average (Ho...	0.0005	0.0005	0.0005	0.0000
					Occurrences	396.0300	353.0000	449.0000	4.2852
					Percent	0.1685	0.1494	0.1907	0.0018
					Total (Hours)	0.1906	0.1691	0.2166	0.0021
				StarvedTime	Average (Ho...	0.2851	0.2520	0.3183	0.0030
					Occurrences	397.0300	354.0000	450.0000	4.2852
					Percent	99.8315	99.8093	99.8506	0.0018
					Total (Hours)	112.8912	111.4932	114.6699	0.1387
		Defects.InputBuffer	Content	NumberInStation	Average	0.0000	0.0000	0.0000	0.0000
					Maximum	1.0000	1.0000	1.0000	0.0000
			EntryQueue	NumberWaiting	Average	0.0000	0.0000	0.0000	0.0000
					Maximum	1.0000	1.0000	1.0000	0.0000
				TimeWaiting	Average (Ho...	0.0000	0.0000	0.0000	0.0000

Figure A.3: Simio experiment results window.

as shown in Figure A.4. The values can then be changed in the properties window. This value in the properties window is only a default value which means that scenarios that were created in the experiment window will not be changed automatically when the value in the properties window is changed. Only new scenarios created after the change in default value will display the new value.

In the validation experiment, the configurations were given an initial capacity combination which remained unchanged during a run. Hence the analyst must physically change the capacity before each different run is performed. Each of the machines has a resource situated behind it. Figure A.5 shows and example of a part feeder's resource and by clicking on the resource, its properties are shown in the properties window.

The property of the resource that is modified is the *initialized* add-on process trigger. Double clicking on *initialized* will take the user to the specific process in the processes window as shown in Figure A.6. If that machine will not be used in the specific run, the analyst will place a *fail* step in the process and specify in



Name	ObjectType	DisplayName
Properties		
PF_Type_1_ProcessingTime	Expression Property	PF_Type_1_ProcessingTime
PF_Type_2_ProcessingTime	Expression Property	PF_Type_2_ProcessingTime
InspectionTime	Expression Property	InspectionTime
Good_RemovalTime	Expression Property	Good_RemovalTime
Defects_RemovalTime	Expression Property	Defects_RemovalTime
PalletPlacingTime	Expression Property	PalletPlacingTime
Conveyor_Speed	Numeric Property	Conveyor_Speed
Pallets_Q1	Numeric Property	Pallets_Q1
Pallets_Q2	Numeric Property	Pallets_Q2
Pallets_Q3	Numeric Property	Pallets_Q3
Pallets_Q4	Numeric Property	Pallets_Q4
Pallets_Q5	Numeric Property	Pallets_Q5
Total_Pallets	Numeric Property	Total_Pallets
Properties (Inherited)		

Properties: PF_Type_1_ProcessingTime (Expression Property)

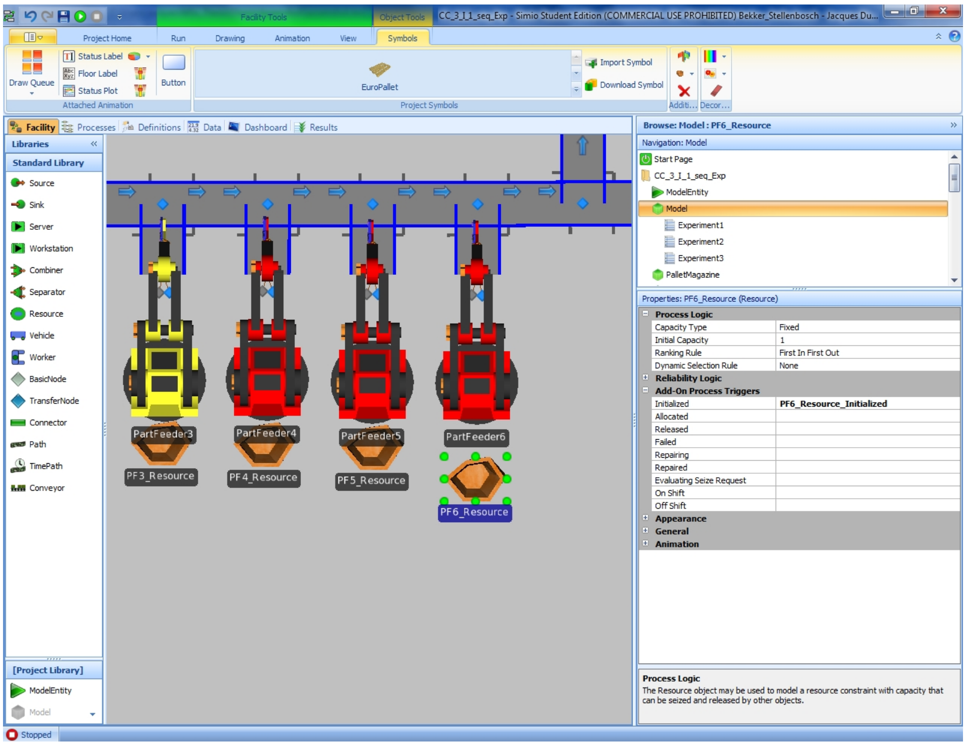
Logic	
Default Value	**Random.Triangular(1,1.1,1.2)**
Switch Property Name	
Switch Condition	Equal
Switch Value	
Foreign States	False
Unit Type	**Time**
Default Units	**Seconds**
Appearance	
Display Name	**PF_Type_1_ProcessingTime**
Category Name	General
Category Expanded	False
General	
Name	**PF_Type_1_ProcessingTime**
Description	
Required Value	True
Visible	True

Logic
An expression can be formed from properties, states, random functions, math functions, and constants using the operators + - * / ** > >= < <= == != && and || with precedence groupings formed using left and right parenthesis (). Parenthesis are

Figure A.4: Machine processing times.

the properties window to fail the particular machine's resource (In this case part feeder 6).

The next control that was used in the validation experiment is the order that had to be completed. Simio can use data sheets to drive the model through time which is accessed by going to the data tab. As shown in Figure A.7 a data table can be imported from Excel by using the *bind to* button in the top ribbon. When the models are running in mixed and batch mode respectively, they use different data sheets which can be found on the CD. State variables that are created in the definitions window, using the states panel, can then be assigned values from these imported data sheets. Refer to the introductory notes of Simio for a guide on how to accomplish this. The order size for each type of Q-frame is assigned to a variable as well as the number of pallets for each type of Q-frame. Note that the number of pallets stay the same during a run for the validation experiment but they do change in the other experiments.



Properties: PF6_Resource (Resource)

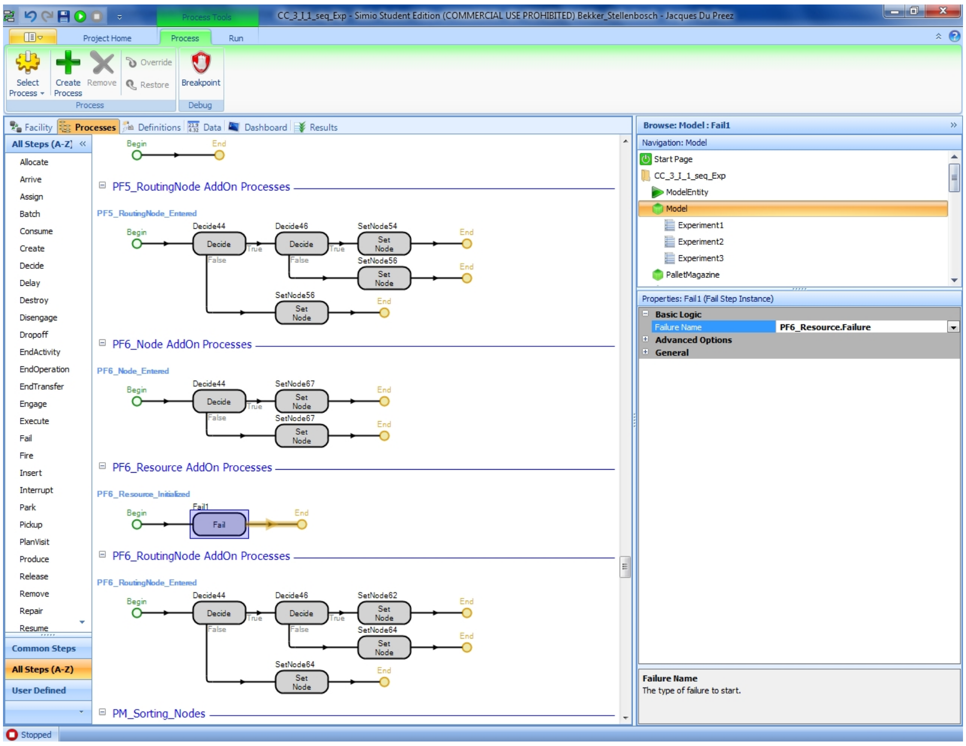
Process Logic	
Capacity Type	Fixed
Initial Capacity	1
Ranking Rule	First In First Out
Dynamic Selection Rule	None
Reliability Logic	
Add-On Process Triggers	
Initialized	**PF6_Resource_Initialized**
Allocated	
Released	
Failed	
Repairing	
Repaired	
Evaluating Seize Request	
On Shift	
Off Shift	
Appearance	
General	
Animation	

Process Logic
The Resource object may be used to model a resource constraint with capacity that can be seized and released by other objects.

Figure A.5: Part feeder resource.

A.2.2 Controls for the Pareto Exhaustive Search Experiment

In the Pareto exhaustive search experiment the number of pallets and the conveyor speeds were changed. In addition to these changes, the capacity was also changed. This resulted in 81 scenarios for each capacity combination. The same orders that were assembled during the validation experiment, were assembled during the Pareto exhaustive search experiment. The only difference being that the number of pallets were not assigned using the data sheet, but rather state variables. The reason for this is that the values in the data sheet cannot be changed while the model is running. Hence the number of pallets were made properties of the model, in the same way the machine processing times were made. The required values for conveyor speed and the number of pallets for each scenario are then manually typed into the experiment design window.



Failure Name: PF6_Resource.Failure

Failure Name
The type of failure to start.

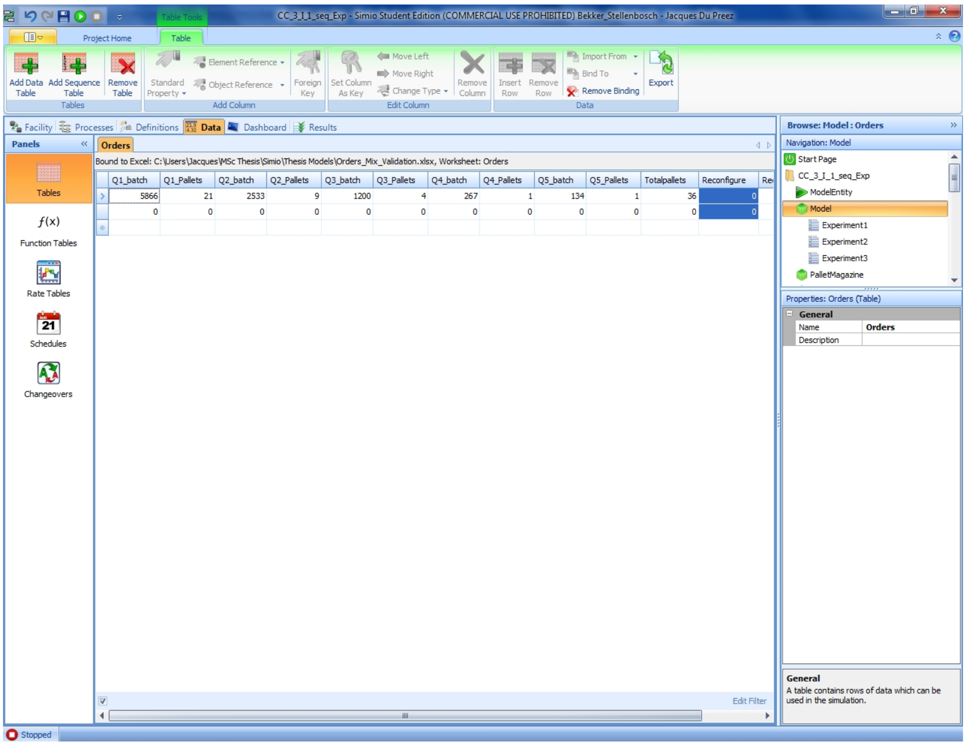
Figure A.6: Add-on process trigger for a part feeder resource.

A.2.3 Controls for the Conveyor Speed Adjustment Experiment

The controls that were changed in the conveyor speed adjustment experiment are exactly the same as the controls in the Pareto exhaustive search experiment. The only difference was that the conveyor speed and the number of pallets were varied in a different way. Refer to *Experiment 2* of the Simio model *CC 2 I1 seq* on the CD for details.

A.2.4 Controls for the Ramp-up Time Experiment

In the ramp-up time experiment, a production scenario was created using a data sheet and a rather complex add-on process. In the production scenario the system undergoes capacity changes automatically once an order is completed. The number of pallets in the system as well as the order sizes are also changed au-



Bound to Excel: C:\Users\Jacques\MSc Thesis\Simio\Thesis Models\Orders_Mix_Validation.xlsx, Worksheet: Orders

Q1_batch	Q1_Pallets	Q2_batch	Q2_Pallets	Q3_batch	Q3_Pallets	Q4_batch	Q4_Pallets	Q5_batch	Q5_Pallets	Totalpallets	Reconfigure	Re
5866	21	2533	9	1200	4	267	1	134	1	36	0	
0	0	0	0	0	0	0	0	0	0	0	0	

General

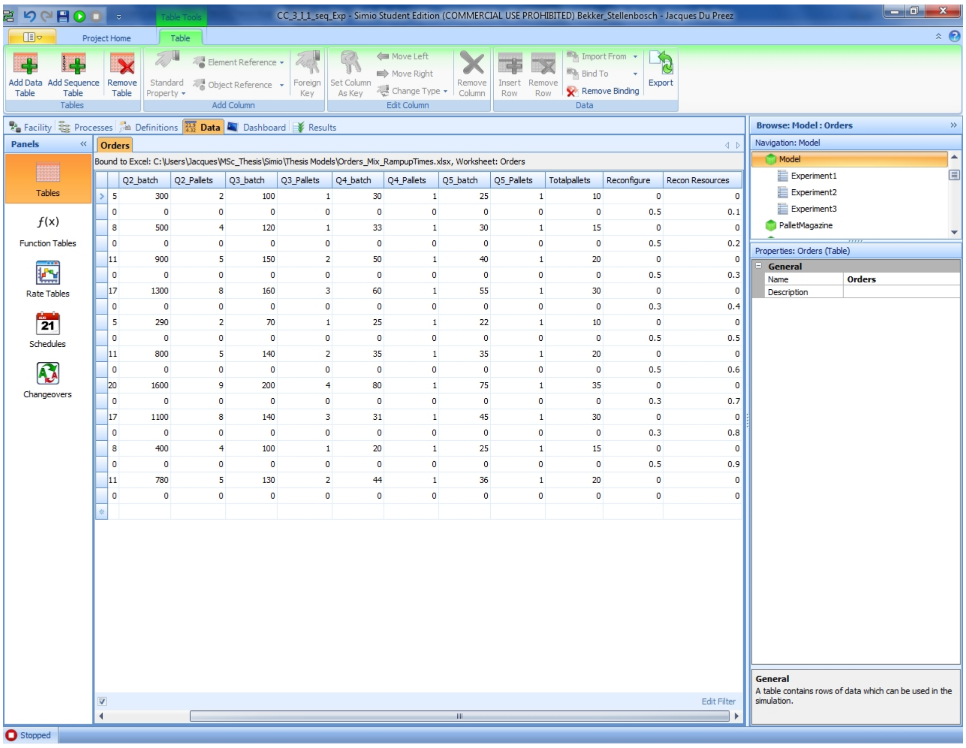
Name	Orders
Description	

General
A table contains rows of data which can be used in the simulation.

Figure A.7: Simio Data table example.

tomatically using the data sheet. An example of this data sheet for when the system is operating in mixed mode can be seen in Figure A.8.

The last two columns *Reconfigure* and *Recon Resources* respectively specify whether the system will increase or decrease capacity and which machines are added or removed. The add-on process that makes the reconfiguration possible can be viewed in the processes tab under the name *reconfiguration*. Unfortunately this process is too large to fit into a screen capture and therefore no figure is provided. The reconfiguration delays and the batch delays are also included in the process *reconfiguration*. These can easily be changed by clicking on the appropriate *delay* step and changing the delay time in the properties window.



Bound to Excel: C:\Users\Jacques\MSc_Thesis\Simio\Thesis Models\Orders_Mix_RampupTimes.xlsx, Worksheet: Orders

	Q2_batch	Q2_Pallets	Q3_batch	Q3_Pallets	Q4_batch	Q4_Pallets	Q5_batch	Q5_Pallets	Totalpallets	Reconfigure	Recon Resources
5	300	2	100	1	30	1	25	1	10	0	0
0	0	0	0	0	0	0	0	0	0	0.5	0.1
8	500	4	120	1	33	1	30	1	15	0	0
0	0	0	0	0	0	0	0	0	0	0.5	0.2
11	900	5	150	2	50	1	40	1	20	0	0
0	0	0	0	0	0	0	0	0	0	0.5	0.3
17	1300	8	160	3	60	1	55	1	30	0	0
0	0	0	0	0	0	0	0	0	0	0.3	0.4
5	290	2	70	1	25	1	22	1	10	0	0
0	0	0	0	0	0	0	0	0	0	0.5	0.5
11	800	5	140	2	35	1	35	1	20	0	0
0	0	0	0	0	0	0	0	0	0	0.5	0.6
20	1600	9	200	4	80	1	75	1	35	0	0
0	0	0	0	0	0	0	0	0	0	0.3	0.7
17	1100	8	140	3	31	1	45	1	30	0	0
0	0	0	0	0	0	0	0	0	0	0.3	0.8
8	400	4	100	1	20	1	25	1	15	0	0
0	0	0	0	0	0	0	0	0	0	0.5	0.9
11	780	5	130	2	44	1	36	1	20	0	0
0	0	0	0	0	0	0	0	0	0	0	0

General

Name	Orders
Description	

General

A table contains rows of data which can be used in the simulation.

Figure A.8: Ramp-up time experiment data sheet example.