UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

DEPARTEMENT BEDRYFS- EN SISTEEMINGENIEURSWESE
DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

FRONT PAGE FOR FINAL PROJECT DOCUMENT (BPJ 420) - 2016	
Information with regards to the mini-dissertation	
Title [e.g. A Contingency framework for the after-sales inventory at Xxxxx Part Distribution Centre]	Minimise loom changeover time at MONN
Author [Last name, Initial(s) e.g. Botha, P.J.]	Nouwens, M.S.
Student number	14075114
Supervisor/s [Last name, Initial(s) e.g. Botha, P.J.]	Joubert, J.W.
Date [(year/month/day) e.g. 2012/10/08]	2017/09/28
Keywords [Provide keywords from your project (for searching purposes). The first word of two words must always have a capital first letter and the rest of the words following must be lower case. In the event of an abbreviation, use as it is known, e.g. Economic recession, ABSA, ERP, Simulation modelling)]	Simulation Modelling, Changeover time reduction, Throughput, Carpet loom, Creeling, Ergonomics.
Abstract [Provide an abstract of the mini-dissertation. An abstract is a short summary of the contents covered in the item.]	MONN is a manufacturer of premium quality and custom woven Axminster carpeting. The process of making a custom woven Axminster carpet involves many processes but the weaving process offered room for improvement in the throughput and changeover time of the 5 looms, which weave the carpets. It was identified that the looms created a bottleneck in the factory making any improvement in loom changeover time valuable. The processes and methods in place when setting up a loom for a new job were unergonomic and slow. Using multiple literature sources that looked into facilities planning and the ergonomics of preparing the looms, new methods improving ergonomics and speed were developed. Two new scenarios were developed which could lead to possible improvements over the current system. In order to assign a monetary value to the different scenarios, AnyLogic was used

	a simulation tool to model the different scenarios. AnyLogic is a simulation program that by utilising the concept of discrete and agent based modelling provides a platform to simulate a process with accuracy. The current processes and systems MONN uses was simulated providing a foundation for the new scenarios for and benchmark to which they can be compared to. To ensure that the base model was realistic; validation and calibration experiments were conducted. Once the based model was completed, the new scenarios could be developed using simulation. Two scenarios were developed where changes in the material handling systems and improvements in general ergonomics were made. An additional scenario was added where staff were relocated to new positions to further improve performance. The results of the scenarios' simulation models were documented, and the monetary values of the simulations were assessed to determine the feasibility. It was found that all proposed scenarios were an improvement over the current system and would that they would pay themselves in 5–8 years contributing R500000 – R1350000 a year to pay themselves off, depending on the scenario.
Category [Enter the category in which the project was done. E.g. Operations Research, Simulation, Logistics…]	Simulation
Sensitive Parts [Indicate if there are sensitive parts that should not be published on UPSpace, e.g. Appendix A and H should not be published.]	N/A
Declaration	
1. I understand what plagiarism is and I am aware of the University's policy in this regard. 2. I declare that this is my own original work 3. Where other people's work has been used (either from a printed source, internet or any other source) this has been carefully acknowledged and referenced in accordance with departmental requirements 4. I have not used another student's past work to hand in as my own 5. I have not allowed and will not allow, anyone to copy my work with the intention of handing it in as his/her own work	
Signature	

Minimise loom changeover time at MONN

Marcus Nouwens : 14075114

September 28, 2017

Executive Summary

MONN is a manufacturer of premium quality and custom woven Axminster carpeting. The process of making a custom woven Axminster carpet involves many processes but the weaving process offered room for improvement in the throughput and changeover time of the 5 looms, which weave the carpets. It was identified that the looms created a bottleneck in the factory making any improvement in loom changeover time valuable.

The processes and methods in place when setting up a loom for a new job were unergonomic and slow. Using multiple literature sources that looked into facilities planning and the ergonomics of preparing the looms, new methods improving ergonomics and speed were developed. Two new scenarios were developed which could lead to possible improvements over the current system. In order to assign a monetary value to the different scenarios, *AnyLogic* was used as a simulation tool to model the different scenarios. *AnyLogic* is a simulation program that by utilising the concept of discrete and agent based modelling provides a platform to simulate a process with accuracy.

The current processes and systems MONN uses was simulated providing a foundation for the new scenarios for and benchmark to which they can be compared to. To ensure that the base model was realistic; validation and calibration experiments were conducted. Once the based model was completed, the new scenarios could be developed using simulation.

Two scenarios were developed where changes in the material handling systems and improvements in general ergonomics were made. An additional scenario was added where staff were relocated to new positions to further improve performance.

The results of the scenarios' simulation models were documented, and the monetary values of the simulations were assessed to determine the feasibility. It was found that all proposed scenarios were an improvement over the current system and would that they would pay themselves in 5–8 years contributing R500 000 − R1 350 000 a year to pay themselves off, depending on the scenario.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

MONN is a family business, where the younger generation has always been involved from an early age. At present the fifth generation of the family is involved in the carpeting industry. Owing to the years of exposure in the company, many prospective projects have been identified to improve the performance of the company. The ideas behind this report have been a work in progress since 2014.

1.1 Background

MONN is a manufacturer of premium-quality and custom-woven Axminster carpets for hospitality applications. Axminster carpet refers to a type of carpet production method used to manufacture a multicolour custom-design, cut pile carpet. Producing an Axminster carpet involves processes ranging from purchasing raw wool to finishing the final carpet. MONN is one of the last manufacturers in the world that manufacture the yarn they use to enable total quality control.

1.2 The process

Raw wool is transformed into yarn where-after it is stored until it requires further processing to fulfil a customer order. The yarn is dyed to a colour selected from a palette of over 600 colours. The number of colours and the amount of yarn per colour are determined by the customer's unique design, colour selection, and order size requirements. These factors that define the order are known as the carpet *specification*.

After dyeing, yarn is then sent in the form of a hank, a coil of loose yarn, to the winding department for further processing. Before the actual carpet is woven the yarn needs to be wound into bobbins. A bobbin is, a cylindrical cone with yarn wound onto it, to be used by the looms at a later stage. These bobbins are assembled and wound by machines called *winders* (see figure B.2). A *loom* (see figure B.4) is a machine that processes yarn and other material to weave a carpet. MONN has five Axminster looms and 12 winders. One of the five looms is only used when the factory is in peak production. Normal operations will see four looms running.

The number of bobbins and the amount of yarn on a select bobbin are determined by the carpet specification. Typically a job can consist of 5 000–18 000 bobbins with anything from 30 meters to 2 000 meters of yarn on any given bobbin. The bobbin has a fixed width of 20 cm and a maximum diameter of 14 cm (see figure 1.1); which is determined by the amount of yarn on the bobbin.

When the bobbins undergo the winding process for production, the exact amount of yarn is measured and wound onto the bobbin. A label is placed on the bobbin, specifying which pin position on a creel the specific bobbin has to be placed, on a specific loom (see figure 1.1). Creels are large frames situated on top of the looms which have pins that hold the bobbins in

Figure 1.1: Two images showing front and side view of a bobbin distinguishing what is meant by width and diameter of bobbin. A typical label can also be seen.

place. Each pin has a unique position code and a pipe that runs from the pin position to the Axminster weaving machine feeding the machine with the yarn from the bobbin (see figure B.3 in appendix).

After label application the bobbins then fall from the winding machine into large bins (see figure B.2 in appendix), which are placed behind the machine. Once these bins are full an employee moves the bin to a storage area. When the full bin is needed, it is transported with a forklift to the specific loom. Different bins are sent to specific sides of the creel of the loom. This can be done because the bins are placed in the correct position behind the winding machines so that the bobbins are partially sorted.

Owing to the fact that the bobbins are grouped in bins, it narrows down the possible positions a bobbin can have. Each bin provides for about 1 000 possible positions that are near each other. This aids the employee when creeling — the manual process of placing bobbins onto the creels — as (s)he knows that the next bobbin has to be placed within a specific section. Once creeling has been completed, the mechanical and electronic set-up of the loom can continue to produce the carpet. The bins in which the bobbins arrive, need to be returned to the winding department. Figure 1.2 shows the current factory layout depicting the process locations.

1.3 The problem

There is always room for improvement in a process, especially one such as carpet manufacturing where the original, outdated methods have been accepted as the norm. A problem area that was immediately apparent was, that when the factory was operating at full capacity, a bottleneck was formed by the Axminster looms which resulted in full bins queuing up before arriving at the looms. This would cause the throughput time of a job being lengthened. The bottleneck was formed because the looms, collectively, could not process the work as quickly as other departments.

Any improvement on the time that a loom spends on a job will result in an decrease in throughput time, as well as an increase in capacity. In Axminster carpet manufacturing an issue often overlooked, is the changeover time of a loom. This is a lengthy process because the loom creels, first require that the remaining yarn from the previous job be removed, then the new yarn is loaded, before the loom is set up. The current creeling methods are responsible for the time spent on creeling. Data based on 42 observations of the changeover times was collected at MONN from June 2016 to December 2016.

MONN's changeover times fit a normal distribution as seen in figure 1.3. There are some bins in the histogram which are outliers; it was assumed that it was due to the small sample set.

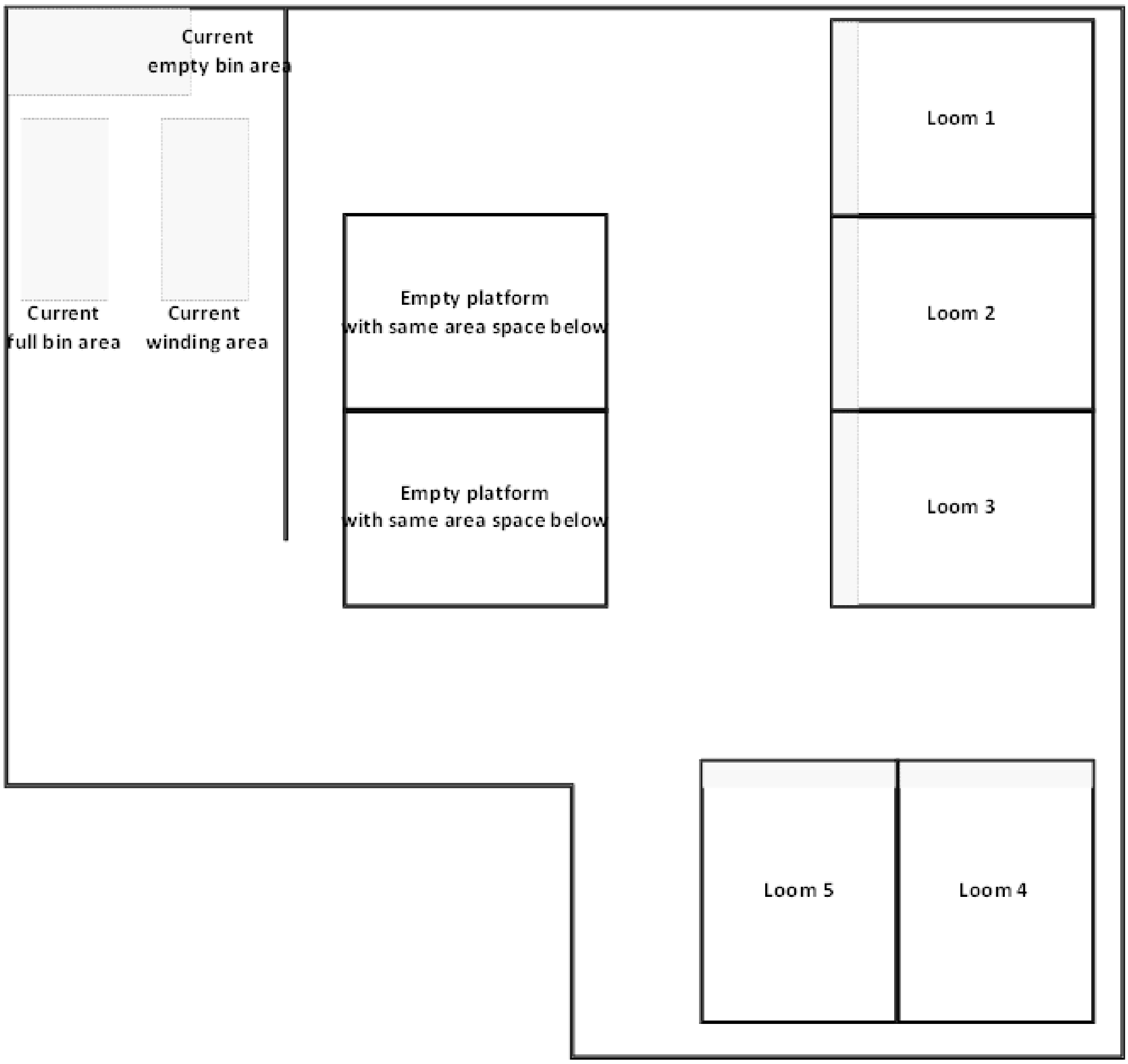


Figure 1.2: Layout of the factory with platform and winding operation area's shown.

The change overtimes are excessive with the mean $\mu = 12.73\ h(hours)$ and standard deviation of $\sigma = 4.192\ h$. According to the latest company costings, a loom costs R920 an hour when standing idle. This means that the loom contributes a mean cost of R11 711.60 changeover costs. *For the smaller orders MONN receives, this cost can comprise of up to 20% of the cost of a order.* Owing to the changeover there is also a loss of capacity, as the looms stand idle during the changeover process. Any improvement in changeover time will result in a increase in capacity. If, for example, there could be a 1% increase in capacity per loom, it would be valued as 5% the cost of a new loom as there are 5 looms in the factory. *Today 5% of a loom value is R977 500.00.*

When the problem at MONN was investigated, it was found that 70% – 80% of the changeover time was when the loom was in the creeling process. A study was done on the creeling method to see why it took so long. When a full bin of bobbins arrived at the creel of the loom, a creeler fetched the bin and moved it to the correct area at the creel. The creeler would then pick up about five to ten bobbins out of the bin, holding them with one arm while leaving the other arm free. Using the free arm, the creeler would take one bobbin at a time, read the identification label, look for the corresponding bobbin position and place it in the creel. This process was repeated until all the bobbins in their arms were in the creel. The creeler then proceeded to

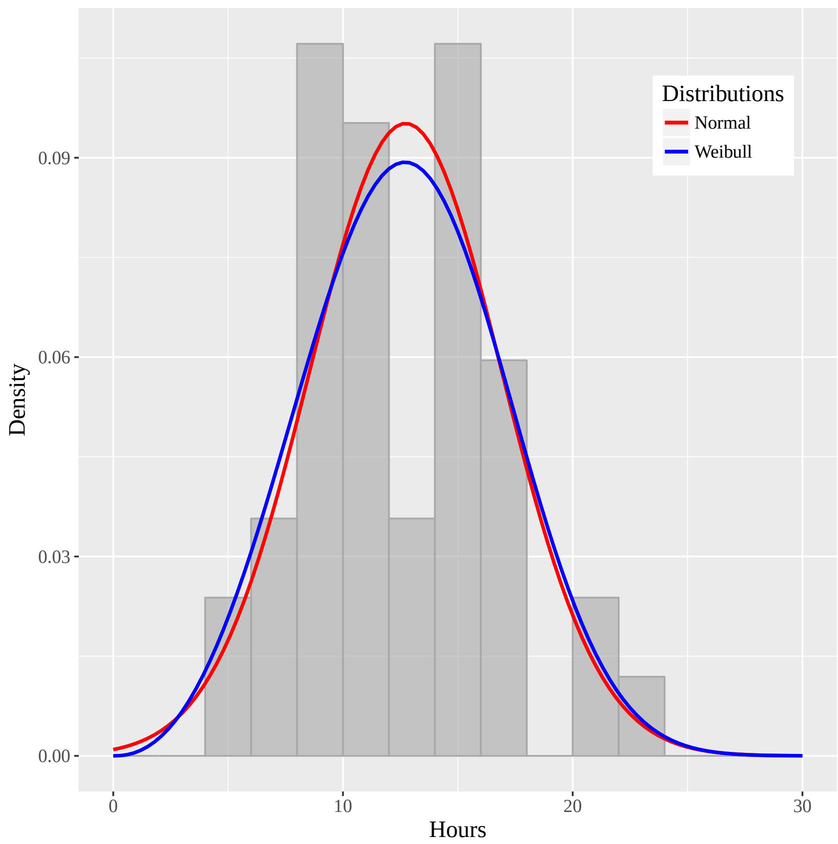


Figure 1.3: Histogram of loom changeover times with two distributions plotted.

collect the next arm full of bobbins out of the bin. When a creeler finished a bin, they moved the empty bin away for collection. If there were a next bin that needed to be creeled the creeler would repeat the above process.

The rate of loading the bobbins was slow at about 20–30 seconds per bobbin, which can be attributed to the fact that MONN use this bins, resulted in the employee having to bend down when picking up the next bobbins. Doing that repetitively, could lead to backache and cramps. Holding bobbins in one arm, while using the other to creel seemed problematic and cumbersome as bobbins easily fell while the employee was moving around. Sometimes bobbins could weigh up to 2kg each with the result that a pack of ten would eventually become very heavy.

The process of reading a bobbin label and then identifying its position in the creel was slow. The bobbins were randomly scattered over an area of 2m$\times$2m with about 150 positions to choose from. Finally, the bins were moved by pushing them along the floor, which is not a problem when the bin is empty. However, it proved to be cumbersome when the bin was full. The bins did not have wheels to aid the creeler and did not provide an appropriate mechanism to either push or pull the bins.

1.4 Research design

It is clear that loom changeover times of MONN can potentially be improved. To do this the methods of creeling need to be significantly revised. As well as the material handling systems from the winding to the creels need to be revised.

The purpose of the project is to develop a new creeling method, along with a appropriate material handling system for the bobbins to get to the creels, potentially resulting in an improvement in the loom changeover time and throughput time. Simulation has been identified as an appropriate method to solve the problem at MONN, especially when there is uncertainty regarding the process that is being modelled. Within the changeover and weaving process there exists uncertainty regarding the job size, the number of bobbins within the job, the winding time per bobbin, and so forth. Throughout the project document, all uncertainties such as these are discussed and dealt with in detail. When using simulation to test new ideas or scenarios, it needs to be supported by facility planning methods so that there is sound reasoning behind the proposed solutions.

The final output of the project comprises of three models: a base simulation model of the current system used by MONN and two alternative scenarios. The two new models have the same new creeling method, but two newly designed material handling systems complementing the new creeling method. The key metrics of the scenario is improved loom changeover times, the throughput time (or job time), and additional capacity as a result of the improvements. The results obtained by the models will be quantified in monetary terms to determine their feasibility.

1.5 Methodology

Firstly, the current system is simulated as a base model providing a foundation for the two alternative scenarios to be compared to. The methodology followed by Law and McComas (2001) on how to develop the base simulation model is discussed below:

Step 1: A basic design of the base model was assembled to understand the data and information requirements better. This design would be in the form of a flow chart. Hereafter, the data and information were collected so that a model of the current system and the two new scenarios could be formulated.

Step 2: Here it was determined whether the based model design, the conceptual design, was consistent with the current process. Here a meeting was held with an industry sponsor to ensure the consistency of the model plans; this was to save time in the future as it would increase the probability that model simulation would provide valid results sooner.

Step 3 and 4: In this phase, the base model was programmed. The base model of the current system was used to test the validity of the model itself, the data and the assumptions made by comparing them to the current system performance at MONN. Sensitivity analyses was done to determine which variables needed to be most accurately modelled.

Step 5: During this phase, the conceptual model developed was tested and analysed. To ensure that the effect of the uncertainties in the simulation did not give a mixture of results when analysing the performance of the system a number of iterations were run so that a mean could be taken there-from. Any major outlying data was investigated and rectified.

Step 6: At this stage, the results of the conceptual model developed were documented.

Next, a new creeling method was determined by investigating better ergonomic and efficient practices. The height of the arrival of the bobbins, the order they were in, and accessibility of the bobbins to the creeler was important.

Once the creeling method was assumed to be good enough, two material handling system alternatives needed to be designed. *The twenty Principles of Material Handling* defined by Kulwiec (1985), were used in the designing of the two new scenarios. The new designs took into account the availability of opportunities within the factory as well as economic feasibility. The new designs were modelled and tested by means of simulation where they were compared to a simulation model of the current methods used by MONN.

As the base model was validated, the new scenario models had a solid foundation to be built on. The new scenario models clearly did not need to be developed from the beginning as much of the data, information and processes remained the same. The methodology followed to formulate new design simulation models was similar to that of the base model in that most steps were the same apart from one, three and four. In step one some additional information and data were collected relevant to the designs. Steps three and four would not require the models to be reprogrammed entirely, but rather just adapted to suit the designs.

The measures taken to evaluate model performance were the mean changeover time, as well as analysis of the distribution of changeover times. In addition to that the time taken for a job to be processed from beginning to end, or job time, was also measured. It was also measured because a system could have an improved changeover time, but a cost in time to process before or after it. Therefore, looking at both changeover time and job time would provide accurate model performance evaluation.

The results of the two scenarios were compared to those of the based model. The model that provided the most economic feasibility in terms of cost and performance was selected.

1.6 Document layout

The methodology followed in finding a solution to the challenge faced by MONN is based on what is suggested by Manson (2006). Manson paves the way by proposing the following research steps:

Awareness of the Problem: This step is where the researcher understands and becomes aware of the problem. This is covered in Chapter 1.

Suggestion phase: The literature of where the design methodology materialised from is reviewed in Chapter 2. Particular reference is made to recommendations made regarding ergonomics in creeling, material handling systems and simulation.

Development: Chapter 3 demonstrates the steps as defined by the design methodology and how the models have been developed. It includes detailed data analysis of all data required so that a base model can be developed. Thereafter it shows how the base model was developed and justified. Chapter 4 shows the design and development of the new scenarios and how they were formulated into models

Evaluation: In Chapter 5 the performance of the new scenarios is measured in loom changeover times. This was achieved by showing the results of the experiments performed with the simulation models.

Conclusion: Chapter 6 concludes the project, recommending a new creeling method with a bobbin handling system shown to be superior.

Chapter 2

Literature review

2.1 Changeover time and its importance

Why is changeover time so important? Goubergen and Landeghem (2002) categorise the different reasons for short changeover times into three main points namely: flexibility improvement, bottleneck-capacities, and cost minimisation.

Flexibility improvement: If you need to produce small lot sizes, then you need to have short changeover times which are appropriate for MONN. This is because some orders can be small and the smaller the order, the larger the effect of a changeover time, as the set-up costs take up a larger portion of the total order cost, as discussed in the problem statement. The size of a job has little to do with setup or changeover time on the loom. The factors that influence changeover time would be number of colour and number of bobbins per colour, both independent of job size.

Bottleneck-capacities: Changeover times need to be minimised to maximise the capacity available for production. Any increased capacity can result in significant capital expenditures as equipment is very expensive. *However, improving changeover time may be a cheaper method to increase capacity.*

Cost minimisation: Any machine has yearly depreciation and maintenance costs, even when only standing idle. This is why it is vital to minimise the time the machine is not actively producing, in other words, revenue.

The authors further state that changeover comprises of four basic elements: the *method*, *technical aspects*, *organisation* and *motivation* of the employees. If any one of these is missing it would result in an increase in changeover time.

Two approaches are suggested to solving changeover time issues. The first approach is to improve equipment design. It is emphasised how important "set-up friendly" equipment is as it has a major impact on changeover times. A set of rules and procedures set out by the authors been followed which have proven themselves over the past ten years in a number of projects. Three of these procedures are appropriate to this project: *simplification*, *standardisation*, and *handling*. The process, in this case creeling, and the equipment that comes with it must be simple, the same for all orders, and easy to execute.

The second approach is to improve the work method. Here it is explained that you cannot expect to minimise changeover time with equipment alone, as the employees still need to be trained to perform the procedure effectively. The changeover time is hindered by the element of method, more specifically the creeling method which is unergonomic and as a result slow. The creeling also lacks technological aspects, like the manner in which the bins are being handled during the creeling process.

Addressing these problems regarding the creeling method will result in better changeovers. "Better changeovers arise by attending to the quality of the changeover." according to Mileham and Owen (1996).

2.2 Creeling

A study done by Luttmann and Jajer (1992), looks into the ergonomics of loading a creel in a textile factory. Their ideas relate to the contributions of Goubergen and Landeghem (2002), in that both articles discuss how to try and improve changeover time. In the article of Luttmann and Jajer (1992), the creeling process is slightly different to the situation at MONN but the fundamentals are identical. The article takes a very scientific approach analysing electric pulses in the muscles to determine which are under the most strain. They conclude by giving recommendations of how the bobbins should arrive at the creels and how they should be loaded. This recommendation along with the suggestions from Goubergen and Landeghem (2002), of using better equipment were taken into account when designing the new creeling method. With a new creeling method the current process of winding and material handling of the bobbins from the winding to the looms needed to be revised to accommodate the new method.

2.3 Material handling

The process of improving a material handling system is well defined by Kulwiec (1985) who includes detailed information on *The twenty Principles of Material Handling*. These principles were followed when designing the alternative material handling systems for MONN.

In the process of establishing alternatives, a paper by Hammer (1990), makes an excellent suggestion of, "Don't automate, Obliterate," that is followed in this project. He makes it clear in a case study with Ford and Mazda that the solution is not just to automate the current process, but to look at which part of the process can be eliminated. He points out that computers must be used to design new solutions instead of just being used to automate.

Tabibzadeh (1989), discusses the how simulation has become an extremely effective tool for material handling system design. He also mentions a number of other papers amounting to the same solution. This suggests that simulation modelling can be used as an evaluation tool of different alternatives to aid in solving the problem at MONN.

2.4 Simulation

Simulation modelling is the reproduction of a dynamic, real world process so that one can depict changes made to the process. According to Biller and Nelson (2002), the premise behind simulation is that the system one is simulating, has uncertainties. If one ignores uncertainty in a system, one can do a very simple calculation, but this creates a false picture. One does not necessarily need a simulation to estimate a new solution, but one cannot plug in mean values and expect to discover the mean performance of the system itself. The more complex the system is, the more appropriate simulation becomes.

In the winding and bobbin handling processes, there are many uncertainties in variables that require a data model to depict what is actually happening. This makes simulation an ideal tool to address the challenge at MONN.

AnyLogic has been used in this project. *AnyLogic* comprise of three simulation types; discrete event based, agent based simulation, and system dynamics.

According to AnyLogic (2017), discrete event modelling can represent a system as a sequence of operations performed on entities. These entities pass through the sequence of operations and have certain attributes that can change throughout the system. Typically when using discrete event modelling many *physical level* details that can be unique to one entity need to be ignored. Discrete event modelling must only be used when the system can naturally be described as a sequence.

Agent-based modelling is appropriate when the modeller needs active entities where the agent defines the entities behaviour. These agents can be put in environments and connected with other agents, exchanging information and running the simulation. For problems in manufacturing agent-based modelling, allows for machinery to react a certain way depending on the agent they are processing. For example, the length of a bobbin can be specified, and the machine will respond to that specific bobbin length.

System dynamics is used for long-term simulation models, combining both discrete and agent-based simulation. This method is used when there is a level of abstraction and it assumes a high level of aggregation of the objects being modelled. This is very often used for strategic simulation models. The simulations done in this project were not of the level of abstraction found when using system dynamics simulation and so would not be used.

When analysing the remaining modelling types, discrete and agent-based modelling, it becomes apparent that the problems MONN is facing can be solved using a combination of the two methods. The system from an incoming job, winding all the way to the loom follows a sequence of operations. However, the entities passing through the system have unique properties. For example, a job has a unique specification such as number of colours, the size of the order and so on. This means that MONN requires a combination of the two and *AnyLogic* provides this option.

In the case of this project, the process from when a job arrives at winding to the looms, was mapped using discrete based modelling. Entities, such as a `JobSpec`, `Worker`, and `Breakdown` job to name but a few are in the form of agent based simulation.

2.4.1 Building the simulation models

Law and McComas (2001) suggest seven steps to building a simulation model. The first step is to "Formulate the Problem". This was done before the simulation modelling started allowing us to move directly to step two of seven which are as follows:

Step 2: Collect information and construct conceptual model. A basic design of the model must be developed for fundamental understanding. Hereafter data must be gathered to specify model parameters and probability distributions. All assumptions, algorithms must be documented. It is also important to collect performance data for the use of model validation in step five.

Step 3: Is the conceptual model accuracy. A meeting must be arranged with the industry sponsor and conceptual model designs must be discussed to ensure the accuracy thereof. If the model is not deemed accurate, there must be a reversal to step two to ensure that the conceptual model designs represent the current system of MONN accurately.

Step 4: program the model. Here one must programme the models on the selected simulation software. A model of the current system should be programmed to support the validity of the conceptual models.

Step 5: Is the programmed model valid. Compare current system model performance measures with the actual performance measures measured in step two for results validation. Parameters that could not be determined with certainty underwent calibration so that the model would produce valid results. It was also important to perform sensitivity analyses to determine which variables needed to be modelled the most accurately.

Step 6: Design, conduct and analyse experiments. For all models, tactical issues such as run length and number of independent runs had to be decided to ensure that the effect of the uncertainties in the simulation did not give a mixture of results and so that accurate performance results could be measured.

Step 7: Document and present the simulation. The final document should include all results along with a detailed description of the of the conceptual models.

Having these steps as a guide all simulation models within this project will follow these steps when being built. This will ensure that the building process of all models are rigorous and consistent throughout the project.

Chapter 3

Model

The base model replicates the process of MONN from when orders begin to be processed at winding, till they are concluded when the loom finishes a job. As stated in the methodology, initially a flowchart (see figure 3.2) was assembled as a basic design for fundamental understanding.

Figure 3.1 offers a description of each of the different shapes used in figure 3.2. Figure 3.2 shows an incoming job being entered into the first process block, namely *Job processing and winder allocation*. Essentially this is the customer's order that has arrived with its dyed yarn which needs to undergo the winding process. Before the yarn can be wound, the winding work required for the job needs to be split between the winders. This is done in the first process block. Once the winders have been allocated winding work, the winding can start and the bins filled.

The winding is an ongoing process, and the full bins proceed to *Storage allocation* where a bin is assigned a space in the bin storage area. Hereafter the bin proceeds to *Bin storage*. Bins that become available for further processing are allocated to the next available loom in *Loom allocation*, upon the loom becoming available the bins advance to the *creeling* process. Once all bins pertaining to the job have been received at the creels and the bobbins loaded, the loom can commence with the *Loom weaving* process. When that is completed the job leaves the weaving department.

The initial basic design determined that the model would be broken up into four main sections, *winding*, *bobbin storage*, *creeling* and finally *weaving the carpet*.

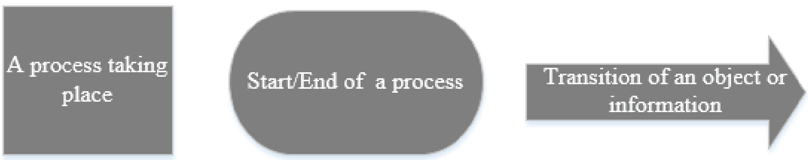


Figure 3.1: Descriptions of shapes used in figure 3.2.

3.1 Winding

Before the process starts, the jobs are created with the information to allow them to be processed. The initial information required for the winding to commence is, namely the *order size*, *number of colours*, *bobbins per colour*, *yarn per bobbin*, *winder parameters* and the frequency of *incoming jobs*. Thereafter the *modelling of winding could commence.*

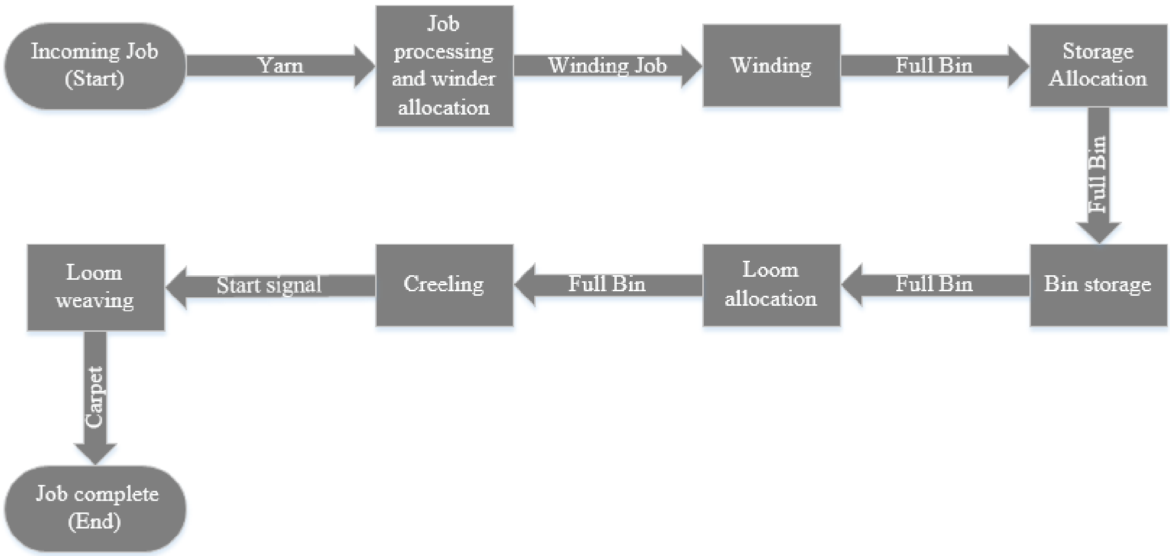


Figure 3.2: A flow chart showing the basic flow of processes with in MONN's Weaving department.

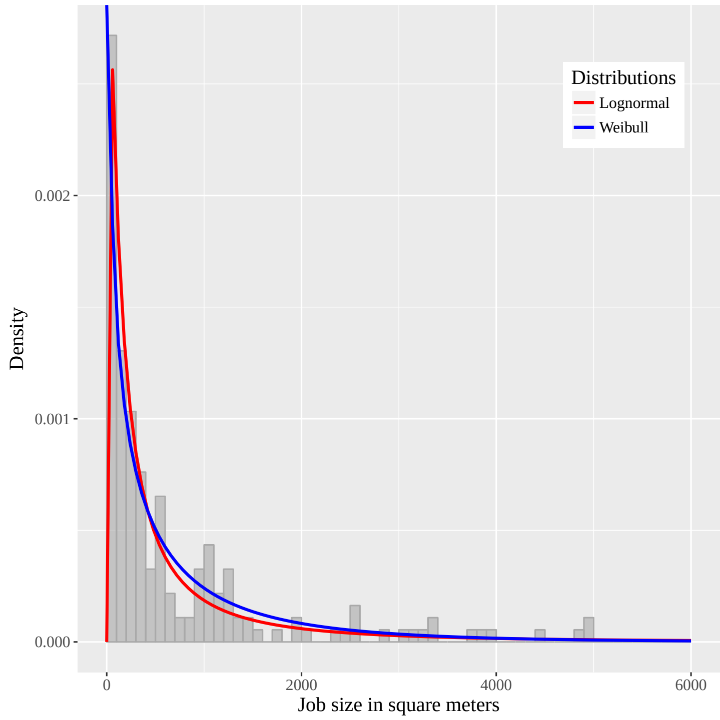


Figure 3.3: Histogram of Order size with two distributions plotted.

3.1.1 Order size

Order size data contained 190 samples from time period January 2016 to January 2017. When the data in `OrderSize.csv` was analysed and plotted into a histogram (see figure 3.3). The the distribution formed matched that of lognormal and Weibull. The lognormal lacked in overall fit and did not represent the data well to the left of the histogram. The data was best estimated to a Weibull distribution with parameters $shape = 0.695$ and $scale = 625.428$. Offhand these values may seem pointless, but these values are needed when trying to replicate the data in the simulation.

3.1.2 Bobbins per colour

The data was taken from 12 jobs and resulted in 109 observations. Data in `BobbinsPerColour.csv` needed to be analysed differently. When looking at the data plotted on a histogram (see figure 3.4) it may seem that the data can fit an exponential curve, but the exponential curve can never be fitted to the data as there is a spike in bin size at both ends of the histogram. To top it all, fitting a distribution to this would be problematic as it could give values that were impossible to achieve. For example it might give a value of 1105 where the maximum possible number of bobbins in a job is 1104. In order to extract an output, *AnyLogic* provided a custom distribution function, for the input of observations. *AnyLogic* then fits a distribution that would output values that correspond accurately. Owing to the uniqueness of the distribution, that seemed an appropriate method to follow.

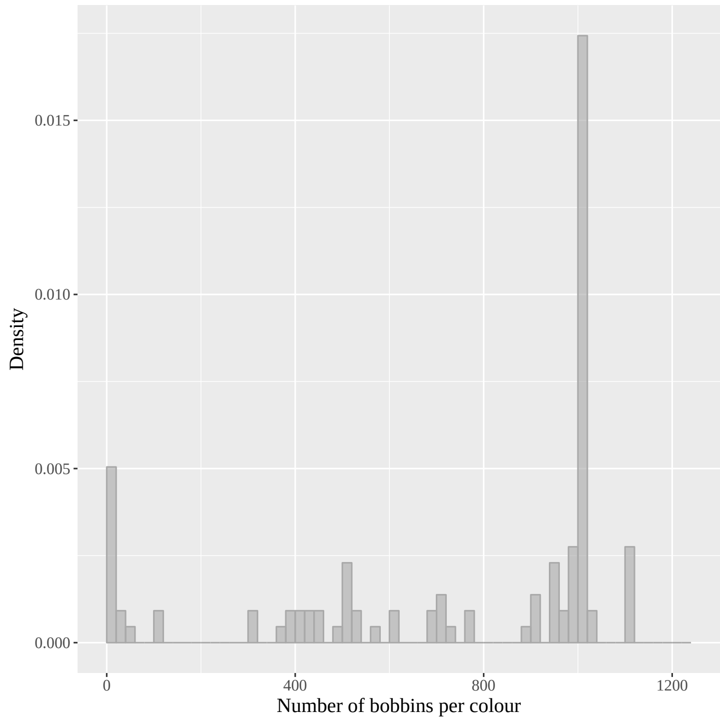


Figure 3.4: Histogram of the number of Bobbins per colour with two distributions plotted.

3.1.3 Number of colours

Next the data in `NumberOfColours.csv`, representing the number of colours in a carpet design, also posed the same problem as seen by number of bobbins per colour where the distribution needs to be followed exactly in order to make it realistic. It can be seen that in the histogram (see figure 3.5) the data possibly follows a Weibull distribution, but there are some bins that do not follow the distribution as they have a lower count than bins surrounding them. This is because customers pay for either two, four, six, eight, or ten colours within the carpet when specifying what they want. If, for example, a customer wants seven colours they will pay the eight-colour price, encouraging the customer to select eight colours. In order to follow this distribution more accurately *AnyLogic* custom distribution function was again utilized.

3.1.4 Yarn per bobbin

To determine the amount of yarn on a specific bobbin, the model uses order size, number of colours per job and number of bobbins per colour. Firstly, it is necessary to determine the

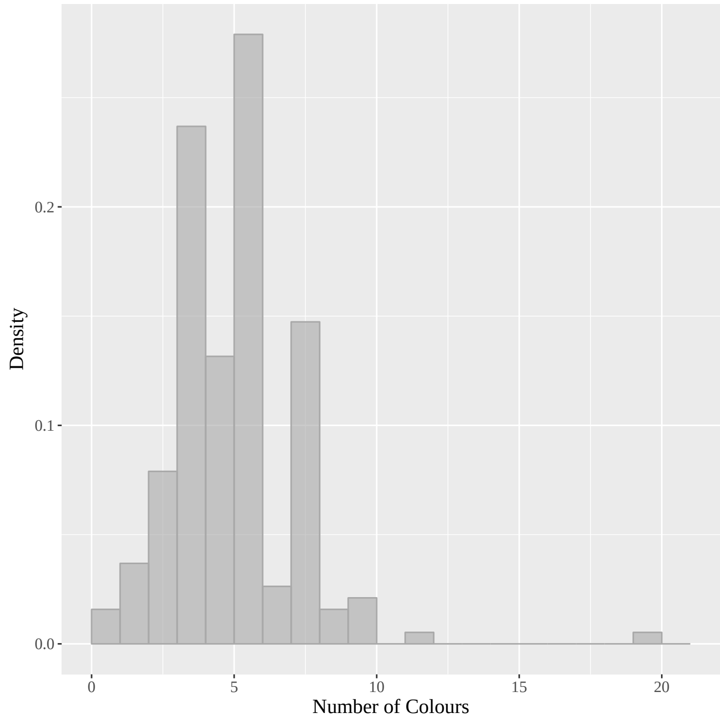


Figure 3.5: Histogram of the number of colours per job with two distributions plotted.

amount of yarn per colour. The order size will indicate how much yarn is required in total. However, it needs to be determined how much of the total yarn, a colour occupies. This is known as colour occupancy, which is seen as a percentage of yarn in a colour vs the total yarn in the carpet.

Yarn per colour is not an important variable to get absolutely correct for these models, because once one has a total number of bobbins and the total amount of yarn per job the winding and creeling time can basically be determined. Although the winders have a set speed they still have to process the same number of bobbins, no matter how much yarn there is per colour. Breakdowns of the winders occur when they switch to a new bobbin, and the number of breakdowns is proportional to the total number of bobbins processed, as a winder breaks downs once in every ten bobbins that are processed. Breakdowns are discussed in more detail in section 3.1.6. However, yarn per bobbin will affect the frequency of bobbin production which determines how often a winder will break down.

As seen in the figures 3.6, 3.7 and 3.8 of typical jobs that have 9,7, and 5 colours in the ordered carpet respectively, there is a relationship between the colour number n and the percentage of colour n's occupancy. The data seems to follow a nice curve, but fitting a distribution will not be correct, as one does not just require a signal data point at a time. The model will need to know the colour occupancy of each colour 1 to n and the use the complete distribution simultaneously. If the colour occupancy of each colour is known then one can determine how much yarn is needed for colour n by multiplying the colour occupancy by multiplying the colour occupancy percentage by the total yarn in the carpet. In order to create usable colour occupancy data, a heuristic approach is used to determine a way to replicate this data.

An approach of weighting the colours according to their *colour number* n, was used. The percentage depicting colour occupancy was determined using the equation 3.1.

$$\text{Colour occupancy of colour } n = \frac{\sum_{n=1}^{n} x}{\sum_{x=1}^{max(n)} \frac{x^2+x}{2}} \tag{3.1}$$

See example of a 9 colour job's colour occupancy in table 3.1.

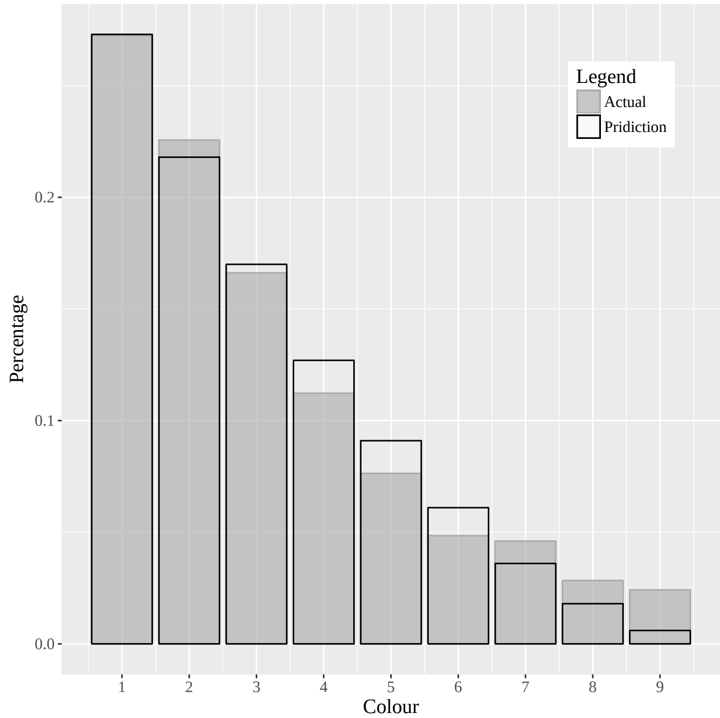


Figure 3.6: Actual job and predicted job yarn occupancy with 9 colours.

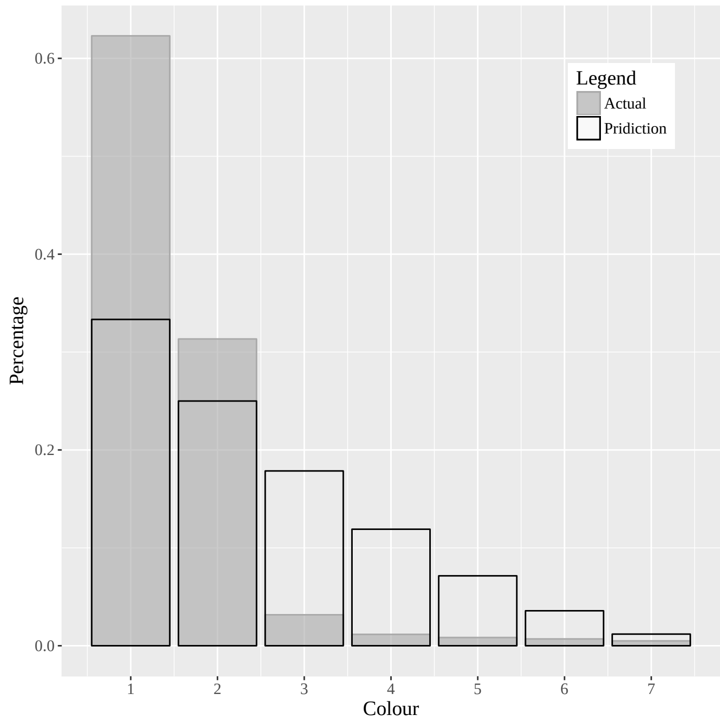


Figure 3.7: Actual job and predicted job yarn occupancy with 7 colours.

Using approach was flexible to number of colours in a job and enabled the model to get a colour occupancy data set for a job with n colours. Figures 3.6, 3.7 and 3.8 show the results generated by this method for a job with 9,7 and 5 colours respectively. One can now compare the results of the prediction method and the what it typically achieved.

When comparing the figures 3.6, 3.7 and 3.8 it can be said that the predicted colour occu-

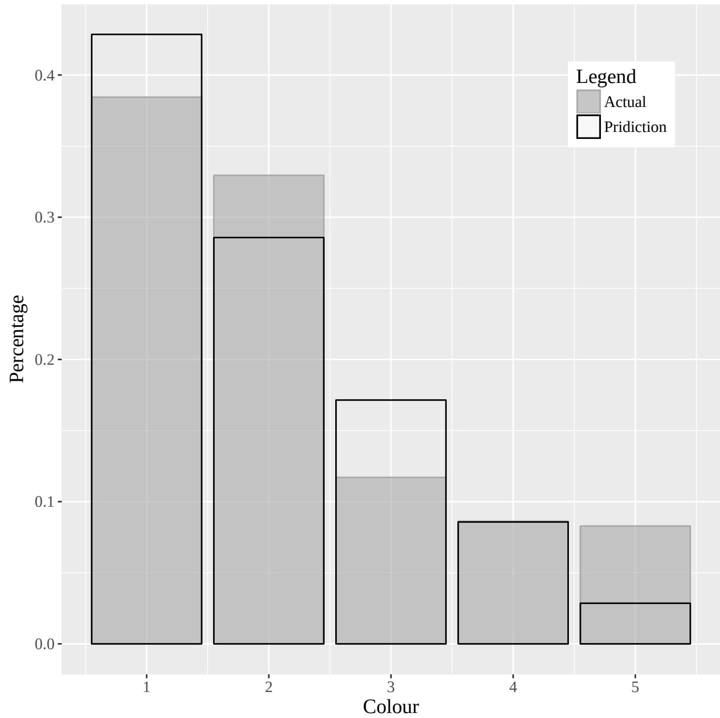


Figure 3.8: Actual job and predicted job yarn occupancy with 7 colours.

Table 3.1: Table showing the colour occupancy of colour n of a 9 colour job.

Colour n	**Colour Occupancy as %**
1	0.6
2	1.8
3	3.6
4	6.1
5	9.1
6	12.7
7	17.0
8	21.8
9	27.3

pancy is close enough to the typical actual colour occupancy. These results are consistent with other jobs as well. Finally, the model could now calculate yarn per colour n using equation 3.2.

$$\text{Yarn Per Colour } n = \text{Colour Occupacy } n \times \text{Total Yarn Per Job} \tag{3.2}$$

It is assumed that the colours with the most bobbins will have the most yarn and colours with fewer bobbins have less yarn. That assumption is consistent with what happens in practice.

Now that yarn per colour has been calculated, it is necessary to determine the amount of yarn there is on each bobbin in each respective colour. To do that individual bobbins of six jobs are analysed with equation 3.1.4. This equation gives a ratio of the yarn on a single bobbin to mean bobbin length in a colour.

$$x = \frac{\text{Bobbin length}}{\text{Mean bobbin length per colour}} \tag{3.3}$$

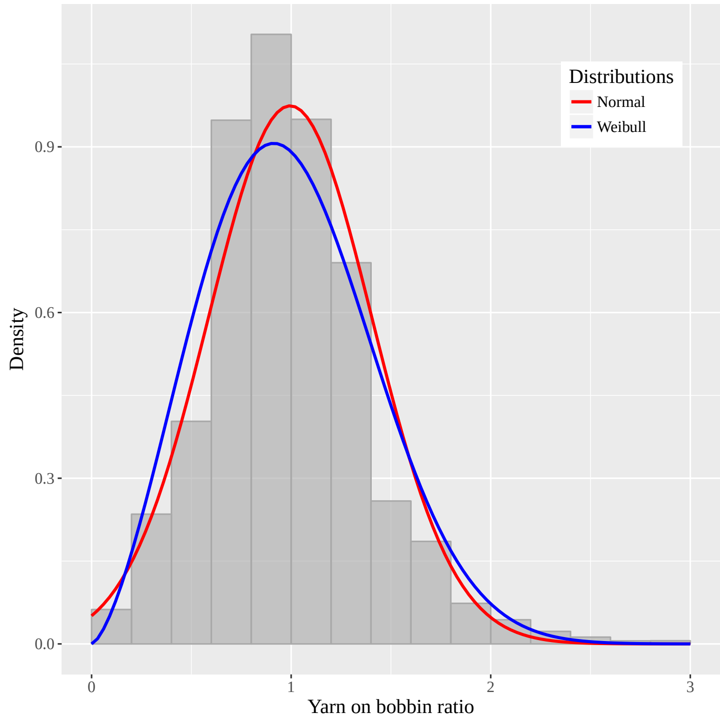


Figure 3.9: Histogram of the the x ratio data with two distributions plotted.

This produced `BobbinLength.csv` with about 42 000 data readings which very appropriately suited a normal distribution (see figure 3.9) with the mean $\mu = 1$ and standard deviation $\sigma = 0.409$. To calculate the amount of yarn on a bobbin the model would use the value received from the distribution x and multiplied it by the mean yarn per bobbin for a colour which was determined by dividing the result of yarn per colour n (see equation 3.2) by bobbins per colour n determined in section 3.1.2.

$$\text{Yarn on a bobbin} = x \times \frac{\text{Yarn per colour}}{\text{Number of Bibbons per colour}} \tag{3.4}$$

3.1.5 Incoming Jobs

Incoming job frequency was determined by looking at the number of orders received between January 2016 and January 2017, and it was found that there were a total of 190 orders received. That determined the number of jobs and the rate at which jobs would enter the models.

3.1.6 Winder parameters

These are values that are used to determine that characteristics of the winding section of the model. Parameters were as follows:

Winder speed: This is the rate, 550 m/min, at which yarn is applied to a bobbin.

Doffing and labelling time: This is a process that occurs when a bobbin has the required amount of yarn applied and the winding process for the bobbin is completed. The process of doffing includes replacing a full bobbin with an empty bobbin, but also applying a label to the bobbin. It takes a winder a total of 21 seconds to complete this sequence.

Breakdown probability and time: This is not a physical breakdown, but an error that occurs during the doffing that forces a shut-down and requires human assistance; this can be a bobbin that is slightly out of place for example. A time study shows this happens once out of ten times and takes the worker about 20 seconds to rectify the error.

Bin capacity: A bin into which bobbins fall, can carry a certain weight of yarn before it is full. This can be translated to the maximum amount of yarn in meters a bin can carry, which is approximately 300 000 meters of yarn.

3.1.7 Modelling of winding

When all the data and information was collected relevant to winding, the based model could commence in design and programming in *AnyLogic*.

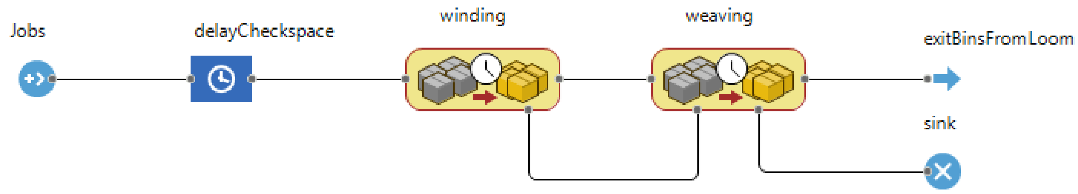


Figure 3.10: Process network for jobs entering the system.

The screenshot (figure 3.10) shows jobs entering the model via source block Jobs. These jobs are in the form of agents called `Jobspecs` that possess the relevant data to allow further simulation. The information is carried in the form of variables and array lists by the agent `Jobspecs` shown in figure (figure 3.11). The variables and array lists are populated when a job is created using the fitted distributions respectively. The number of items in an array list is determined by the number of colours in a job.

Current practice is that before the job is processed in the winding department, or in the case of the model the WINDING module, there are no more than four jobs in full bin storage. This is done to avoid congestion and maintain order in the bin storage space area. This concept is covered in more depth in section 3.2. For now, it is important to know that this is controlled by using a delay process block called delayCheckspace (figure 3.10), which only releases the jobs if the criteria is met.

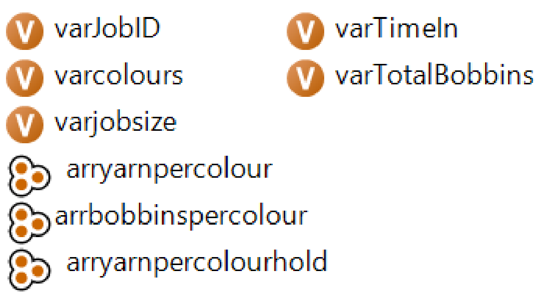


Figure 3.11: The variables stored by agent `Jobspecs`.

Once the jobs are in the WINDING module, they are processed into individual jobs for WINDING and enter via source block WinderJobs (figure 3.13) as agents called `WinderJob`. The agent's variables are populated according to the information received from the `Jobspec` agent. The bobbins within a colour that need to be assembled, are divided between two winders. This is the current practice at MONN and this is how the winders have been set up. For example, if one colour needs 1000 bobbins, each winder would wind 500 bobbins.

Once the agent `WinderJob` is created and populated with information, the `WinderJobs` are allocated to the winder machines depending on availability. If a job had ten colours, it would

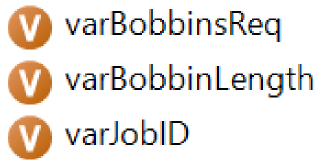


Figure 3.12: The variables stored by agent `WinderJob`.

result in there being 20 winder jobs to complete. With there only being 12 winders eight winding jobs will need to wait until winders become available to process them further. The process of managing which winders get which `WinderJobs` when is done by a process block that was developed called JobAllocator (figure 3.13).

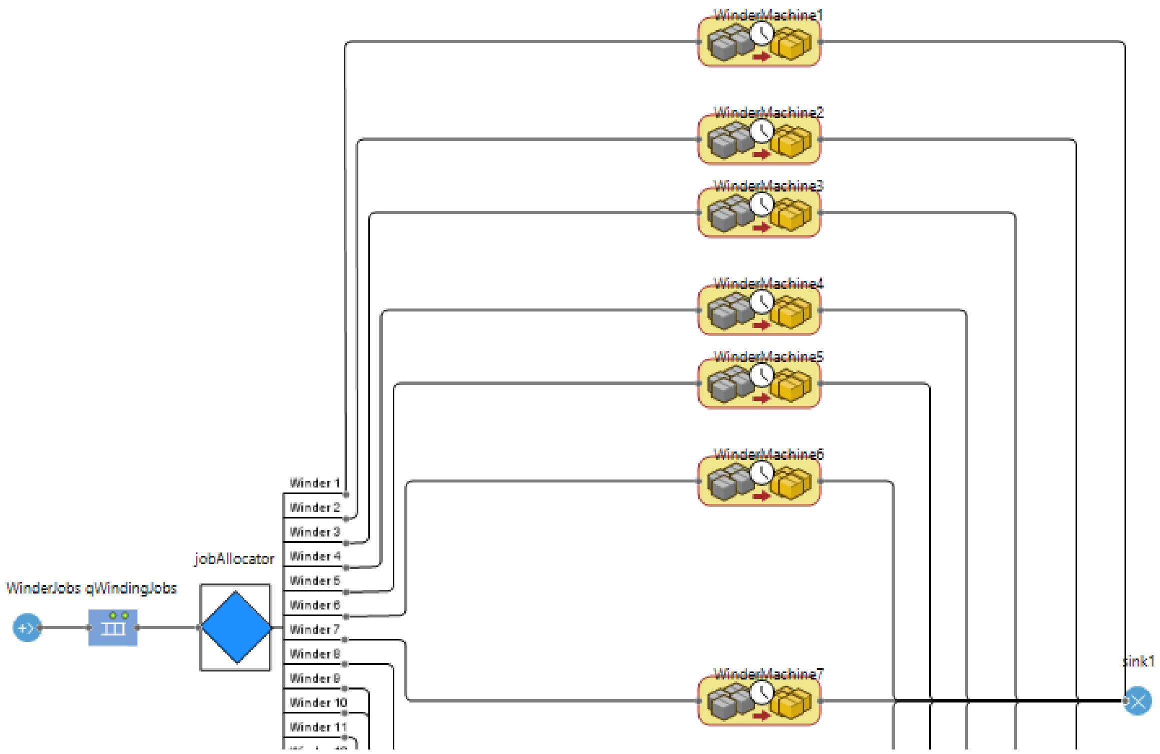


Figure 3.13: Process network for the winding showing 7 out of 12 winders.

When the all the `WinderJobs` agents have been created for a particular `Jobspec` agent and the `Jobspec` agent is released, it will proceed out of the Winding module and back into the main module. There-after the `Jobspec` agent, proceeds into the Weaving module. When a WinderMachine module receives the `WinderJob` agent, the winder's state is defined by a state chart (figure 3.14).

A WinderMachine module needs to communicate to the Winding module. The winder needs to have the ability to call for a worker to either execute a start-up procedure, breakdown procedure, or to fetch an empty bin to be filled. This communication is done with arrays as well as sending messages to and from the WindingMachine module and the Winding module. The worker will execute the requests in the order they are received. When the request is completed a message is sent back to the winder informing it of its current status. As can be seen by the state chart of the WinderMachine module, a winder can only process a `WinderJob` if there is

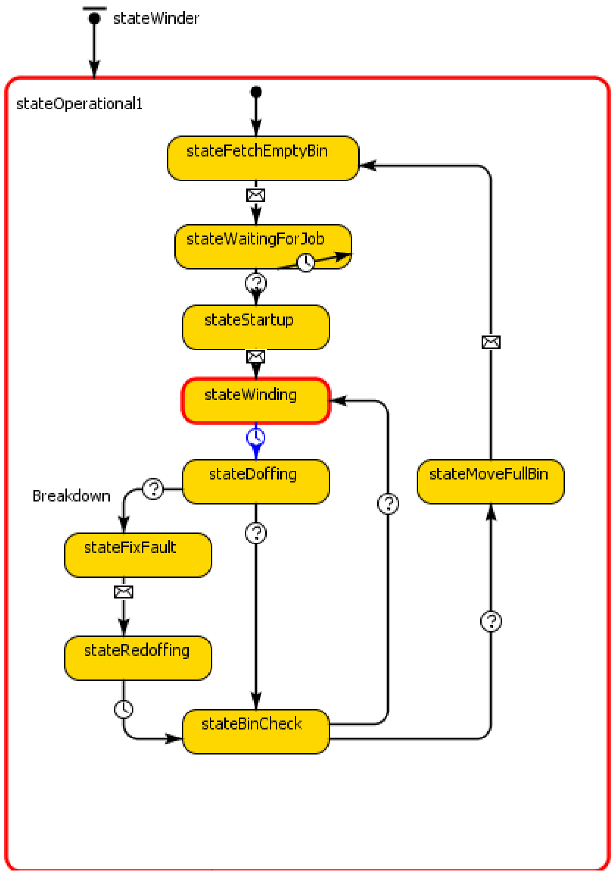


Figure 3.14: Winder statechart used in the model.

an empty bin in place in which the bobbins will fall in. As discussed above the winder will call for a new bin.

To control the bins realistically as it is done in practice, the model sees the bin as an agent who hosts important variables and information (figure 3.15). Once confirmed there is an empty bin in place, the winder then calls the worker for a start-up where-after the winder will begin to complete its job. The information for the winder is acquired through the incoming agent `WinderJob`. The time taken to apply yarn on a bobbin is determined bobbin length received (see section 3.1.4) and winder speed.

Figure 3.15: Variables and information within agent `Bin`.

As discussed in section 3.1.6 on the completion of every bobbin, the winder executes the doffing and labelling process where there is a probability that there may be a breakdown. In the event of a breakdown, the WinderMachine will call for an operator to fix the problem and restart the doffing process.

There are two cases when a bin with bobbins is released from the winder. One if the bin has reached its maximum capacity and needs to be replaced for the `WinderJob` to continue; the other is where the number of bobbins completed is equal to that which was required by the WinderJob. When any of the above two conditions are met, the WinderMachine will release the now populated bin agent back into the Winding module. The winding module will ensure the full bin is collected by the operator from the winder, using process blocks as seen in 3.16 and moved to the storage area. All information is input into bin agent so that the looms can later identify it and all its unique properties, for example number of bobbins.

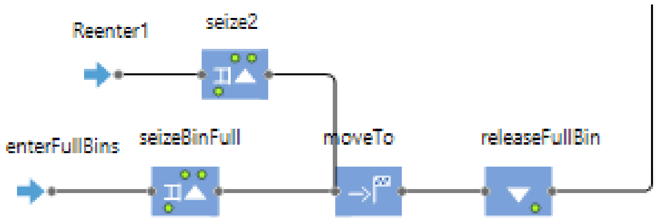


Figure 3.16: Process blocks responsible for making operator fetch full bins.

When the number of bobbins complete is equal to that which was required by the WinderJob the winder will stand idle until a new WinderJob arrives.

3.2 Bobbin Storage

Once the bin arrives at the storage area, it will leave the Winding module and proceed to the Weaving module. The bin storage area space is limited and cannot host more than four jobs at once as the area will become too congested if the value exceeds four. This is managed by delayCheckSpace in the main module (figure 3.10).

When a Jobspecs agent enters the weaving module (figure 3.10), the model will allocate the job to the next available loom and will transfer all the information the loom requires from Jobspec agent to the available Loom module. Once a job has been assigned there is no longer a need for Jobspec, and it is sent back to the main module and then discarded.

The bins received from the winding module will enter into a process block called loomJobAllocator (figure 3.17). Within loomJobAllocator the bins will stand and wait until they are needed. loomJobAllocator will continuously check for bins in the storage area whose varJobID matches the varJobID assigned to a loom. Once loomJobAllocator finds a match(s) the bin agent(s) is/are sent through the network to the relevant Loom module. If there are no bins with matching IDs the loomJobAllocator will wait until it finds a match before sending a bin to the loom.

In order to account for the policy that MONN uses where one loom is only used when the factory is busy, the loomJobAllocator will only assign work to the loom when there are more than four jobs in delayCheckspace found in the main module (figure 3.10).

3.3 Loom Module

Much like a WindingMachine, the state of a Loom is determined by a stage chart (figure 3.18).

When a bin agent arrives in the Loom module, the loom calls the forklift to collect the bin(s) from storage and bring it(them) to the loom. At this stage, the creeling process can commence, and once the bins arrive at the loom, the loom calls for creelers to load the bobbins in the bins onto the creels.

The most important and fundamental data from the agent bin that is used is varJobID as discussed in section 3.2, varBobbinsInBin, and varWinderActive. The varBobbinsInBin tells the loom module how many bobbins there are in a bin and as a result how long a creeler must spend unloading bobbins and creeling them. The time taken to creel the bin is a function of

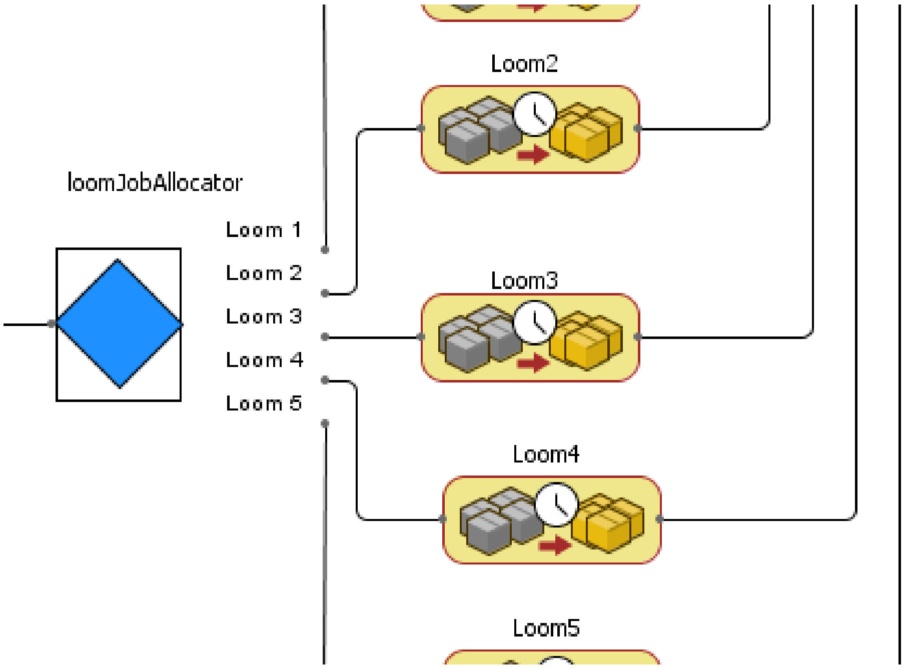


Figure 3.17: LoomJobAllocator in network with the 5 looms.

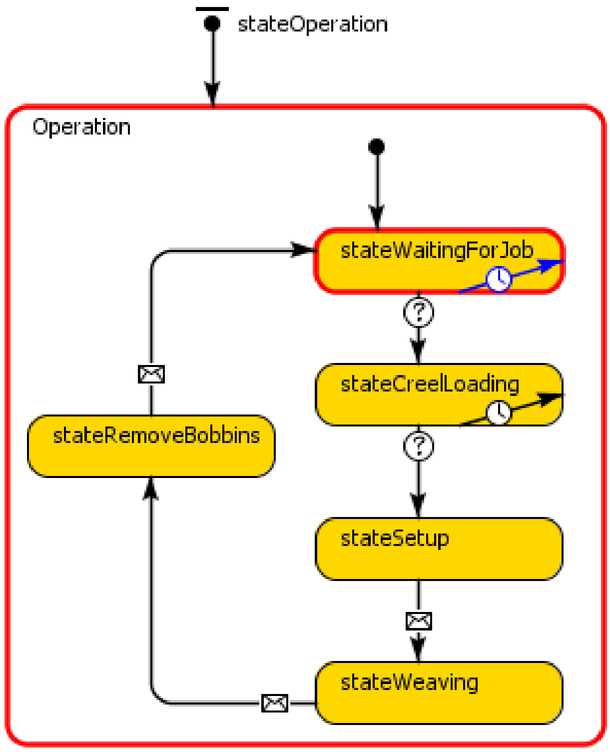


Figure 3.18: State chart that determines the state of the Loom.

creeling rate and number of bobbins in the bin. These are accessed via parCreeling rate and varBobbinsInBin.

varWinderActive is a boolean variable that determines if the bin is the last bin the winder filled for a specific job ID. If the value is false, it means the winder concluded its `WinderJob` on the bin and if true it means that there are more bins to come from the winder. Counting the number of bins that have been creeled that where varWinderActive=FALSE allows the model to know how far the creeling is. As soon as the number of false counts is equal to double the number of colours, it means that all the bins for the job have arrived at the creels. The reason why it is double is that the there are two winders allocated per colour as discussed in section 3.1.7.

Once a bin has been emptied by the creeler the bin goes to one of two places. The bin can go back to the winding storage area for empty bins waiting to be used again, where the LOOM will call the forklift to fetch the empty bin. The bin can alternatively remain at the loom so that when the job has concluded the remainder of the bobbins can be loaded into it. The LOOM ensures that there is one empty bin per colour. The model will make sure that there are no more than 20 bins on a creel to avoid congestion.

Once the creeling has been completed, the loom calls for a loom operator to start the setup process. This is done before weaving the job and takes three hours to complete. There after the weaving of the carpet can start.

The loom module requires some data inputs and parameters to process the job further. The module receives much of the data from the agent `Jobspecs` and from agent bin. The data needed by the loom from the `Jobspecs` agent is the varJobID, varNumberOfColours, varTimeIN and varJobsize.

VarJobsize works together with varloomefficiency (section 3.3.1) and parLoomSpeed to determine how long a loom must weave on a job. ParLoomSpeed is given as $33.6m^2/h$, meaning the machine would produced $33.6m^2$ every hour at 100 percent efficiency.

3.3.1 Loom efficiencies

The loom efficiencies achieved per job were analysed from data in `LoomEff.csv` containing 35 samples. When looking at data such as efficiencies, it is very common that they are normally distributed. Looking at the histogram plot (see figure 3.19) below it can be seen that a normal distribution has a better fit than that of a Weibull, seeming to fit more appropriately with a few bins above or below. If there were a larger sample set available, the outliers would most likely be rectified. The relevant parameters are, mean $\mu = 64.8$ standard deviation $\sigma = 10.414$.

When the weaving job has been concluded the LOOM module calls for one creeler per colour to empty the creels and load the bins with the empty bobbins. This process takes 3 h per colour and requires one worker to execute the process. When the creels have been emptied, and the bins filled, the loom will call for a forklift to take the bins back to the winding. The loom will then wait for its next job.

3.4 Layout of model

Figure 3.20 shows a screen shot of the base model layout while running an iteration. The figure shows that the winders are all processing jobs and bins are being filled. One winder shows a yellow dot meaning it is calling for a worker. All other winders are in winding state with a green dot. The forklift is currently transporting a full bin of bobbins to a loom. There are two looms in production indicated by green dots and two looms in set-up state with yellow dots. One loom is standing idle indicated by a red dot.

3.5 Scheduling

The model contains some resources that need to follow a schedule. MONN works eight hours shifts with one hour break. To accurately simulate the work place, the workers are required to be able to stop their current task at the end of a shift and continue the next shift. *AnyLogic* allows for resources to do task pre-emption, enabling them to leave and later pursue their jobs.

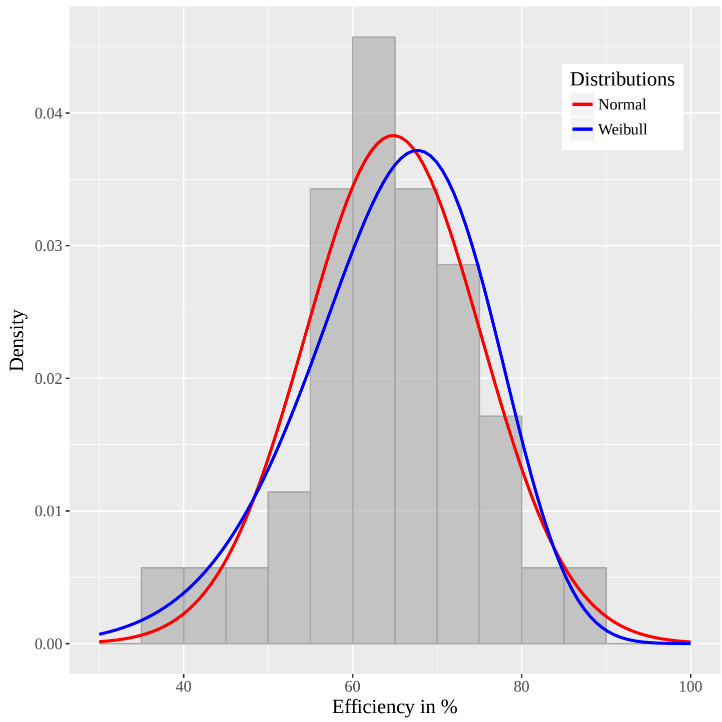


Figure 3.19: Histogram of loom efficiencies with two distributions plotted.

For example, if a creeler is only half way with creeling a bin (s)he will leave the bin where it is at the end of the shift and continue the next day.

This is achieved by inserting a schedule that the factory will follow. The following process block topology, allows for a resource busy with a job, to pre-empt it and continue the next day see (figure 3.16).

It is important to ensure that it is not only the resource units that can pre-empt a task but also a machine such as a winder and a loom. The model was designed in such a way that the modules such as winding machines and looms would hold while there were no operators present.

3.6 Experiment for results

To be able to extract information so that the model performance could be evaluated an experiment was developed. The data required from the model was the mean changeover time of the looms and mean job time (see section 1.5).

The model determined the changeover time by taking the difference between the starting time of the creeling process and the time at which the loom began to weave a job. To determine job time, the time the job entered the model was recorded. That value was subtracted from the time that the loom completed weaving the job and was done for every job. The data was collected and posted to a data set. The model runs for the virtual period of one year, and stores the changeover times and job times that had taken place during that year. It was decided that the model would be run for one year as the company assesses its own performance annually.

To create results that were more accurate and truely replicated model performance, multiple instances needed to be run. Owing to computational constraints, it was decided that 100 iterations would be run. The mean changeover time and mean job time of every iteration were extracted and published as a text file. In order to make the results comparable to other models, a fixed seed number was given to every iteration. For example iteration 56 would run a seed value of 56.

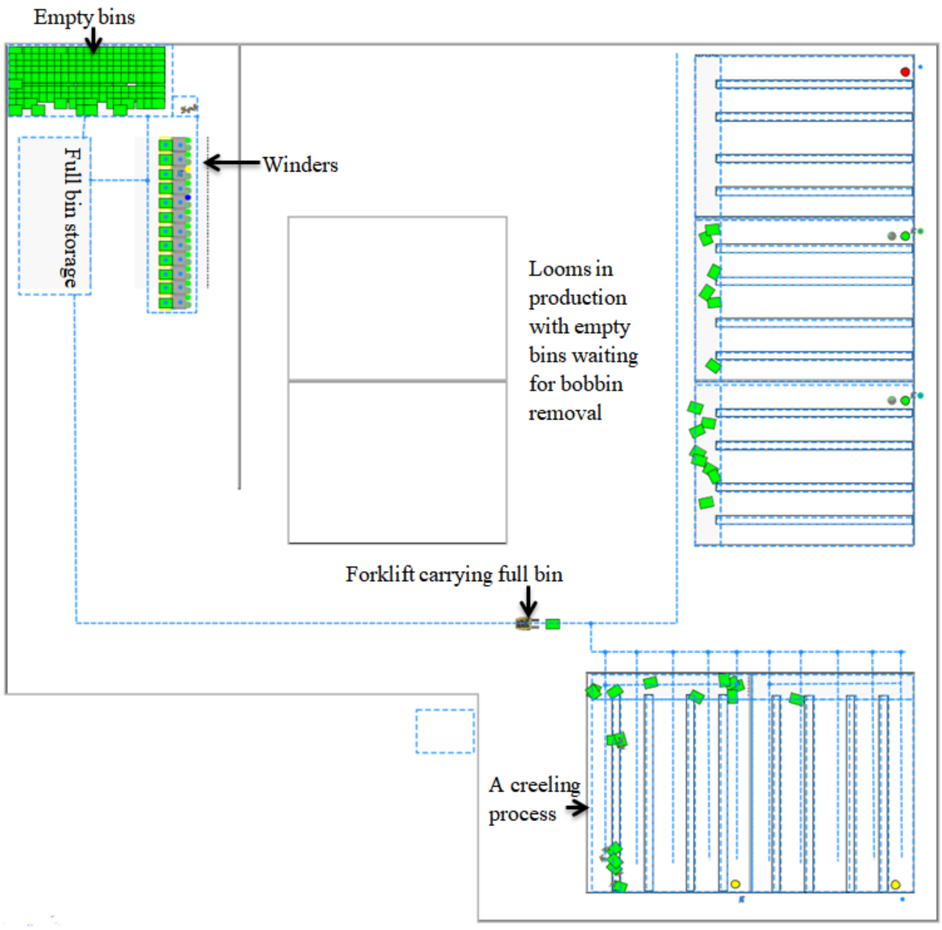


Figure 3.20: A screen shot of base model layout while running an iteration.

3.7 Calibration for validation

If the distribution of changeover times of both a typical iteration of the base model and of the current system at MONN proved to be similar, with the same inputs, the model would be deemed valid and would need to achieve a distribution as *Actual* in figure 3.21.

All the input data modelled was cross validated when establishing their distributions. There were also fixed parameters that were well defined and could not be altered. Some examples of these parameters were: number of winder operators, winder speed, the number of creelers, and loom weaving speed.

There were, however, some parameters that could not be input with certainty. These are parameters where a range of values could be defined instead of one fixed value. Parameters with this characteristic were identified as: walking pace, the number of jobs waiting till the extra loom is activated, the number of jobs held in bin storage area and creeling rate per 1104 bobbins.

To determine what these parameter values should be the model underwent a calibration process. Calibration is where the model runs an iteration for every single parameter configuration possible. Table 3.2 shows the parameters manipulated, their intervals, and the results of the calibration process.

That meant that there were 150 unique possible iterations. The changeover time of each iteration was stored to a text file with the configurations of that iteration. With this text file, it could be seen which combination produced which result. For the results to be more accurate,

Table 3.2: Table showing the parmaters manipulated and the intervals there of in calibration.

Parameters	Unit of measure	Min	Max	Step size	Calibration result
Walking pace	m/s	1.5	2.0	0.1	**1.9**
Jobs queuing till extra loom activation	Number of jobs	6	8	1	**6**
Jobs in bin storage area	Number of jobs	4	5	1	**4**
Creeling rate	h	7	8	0.25	**7.5**
Mean changeover time achieved: 12.707 h					

each configuration was run ten times with seed numbers one to ten according to the iteration number. The mean of the ten mean changeover times for one configuration was taken as the basis to be compared to actual data. The configuration where the mean of the ten mean changeover times was the closest to the actual mean, was selected.

All the newly established parameter values were realistic, which further validated the model. When running individual iterations, the distribution formed by the changeover times of the looms was compared to that of the actual distribution. Figure 3.21 presents an overlay of the actual changeover time of a typical iteration. The distribution produced by the model is similar. However, the model seems to have a tail of changeovers to the right of the distribution. This is due to the creeling sometimes starting before the winding of the job has been completed, and so the creeling process waits for the winders to complete their work. It increases changeover time and creates these results, but it only happens when the factory is quiet.

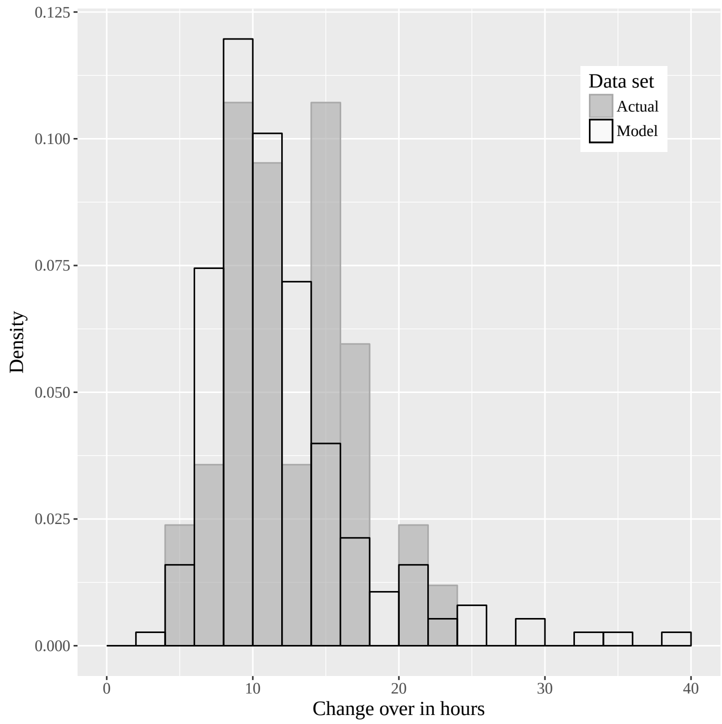


Figure 3.21: Overlaying histograms of the actual data and modelled data.

The *Actual* data does not reflect this because of rather small number of 42 samples. If the data was collected when the factory was busy, it would not have the tails because the there would always be a full job waiting to be loaded on to the loom. If a data set was taken when there are not many jobs in the factory, the tail will start appearing as that happens when the looms stand idle with no work to process.

This explanation made the model valid as the results were constant and reproducible with the actual data. That provided a firm foundation for scenario development and programming.

Chapter 4

Scenario development and Programming

4.1 Scenario development

When considering possible solutions, the problem must be perfectly clear. The definition in section 1.3 highlights that the main culprit in the changeover time is creeling. As mentioned in section 2.2, (Luttmann and Jajer, 1992) has done detailed analysis of the ergonomics in creeling. The study is not for the weaving process such as MONN's, but another type of weaving process. A major difference is that there is not necessarily a fixed position that a bobbin needs to go, in the process studied. However, this does not make their findings unusable, because the process of loading the creels remains the same; there is just not bobbin identification that takes place.

The recommendation was that the bobbins should always be at hip level to reduce the spinal load on the employee. That would mean that the bobbins would be easily accessible to the employee. Further, it was also said that bobbins should arrive in a horizontal position with the appropriate spacing between them. That would make a single bobbin easy to grasp with one hand.

If those two factors could be achieved the creeler would only need to use one arm when loading the creels. Bearing that in mind, the creeler would still need to bend down when the bobbins required, were positioned near the floor. Very little could be done about that as that was how the creels were designed; having all the bobbin positions at an accessible height would require the creels to be larger and taking up more space.

With these recommendations in mind investigation was done to find solutions that could replace the large bins that would still provide an ergonomic, creeling environment. 18th March 2016 Sharks vs Bulls drew 16 all at a rugby game where people selling cold drinks used crates where the bottles were packed upright and the crates were carried by using a sling. That made all the cold drinks within easy reach. If one imagined the bottles being bobbins, crates seemed to be a brilliant solution, although the bottles were vertical and not horizontal. Crates of different kinds were investigated as they were easy to handle and cheap to purchase. Very soon it was found that the standard, bread crate proved to be a suitable candidate.

The bread crates had the ideal dimensions, 690mm × 600mm, to fit eight bobbins that have the maximum possible bobbin diameter of 140mm (figure 4.1). Bread crates also have the advantage that they can be stacked into stacks of 10; meaning you could store 80 bobbins per crate stack; where a stack is 1.5 meters high. The current bin B.1, with dimensions 1200mm × 800mm × 600mm, host anywhere from 200–600 bobbins so a bin. Therefore the bin utilises space more efficiently however as mentioned above it has its disadvantages.

It was assumed that the bread crates would be used. However, it needed to be ensured that it was going to be an improvement on the bins for both the creelers and changeover time. The crates required to be easily moveable around the creels. For that, a simple trolley could

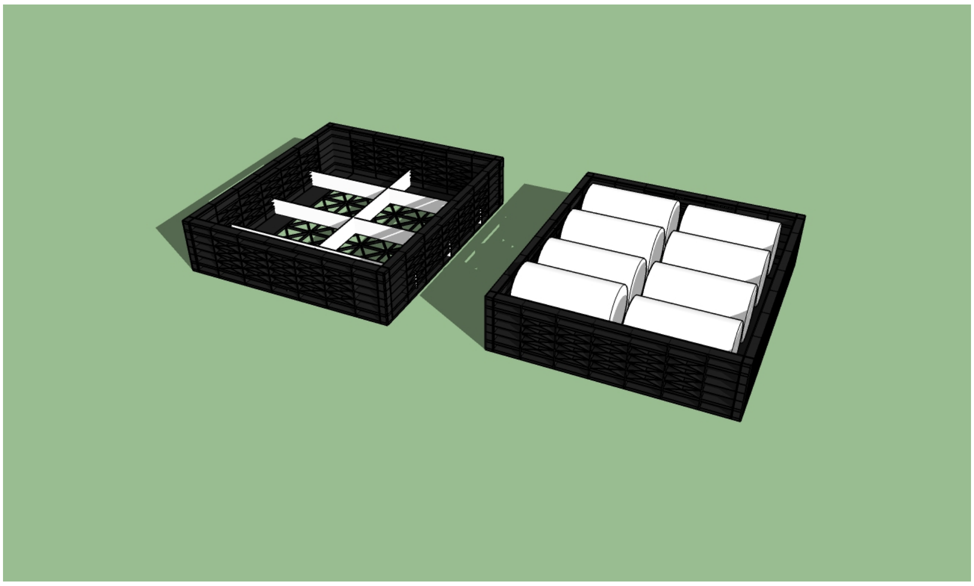


Figure 4.1: 3D model of a crate with inserted dividers (left) and a crate filled with bobbins (right).

be developed that picked up a crate stack to make it manoeuvrable. Figure 4.2 shows the conceptual designs of the trolley.

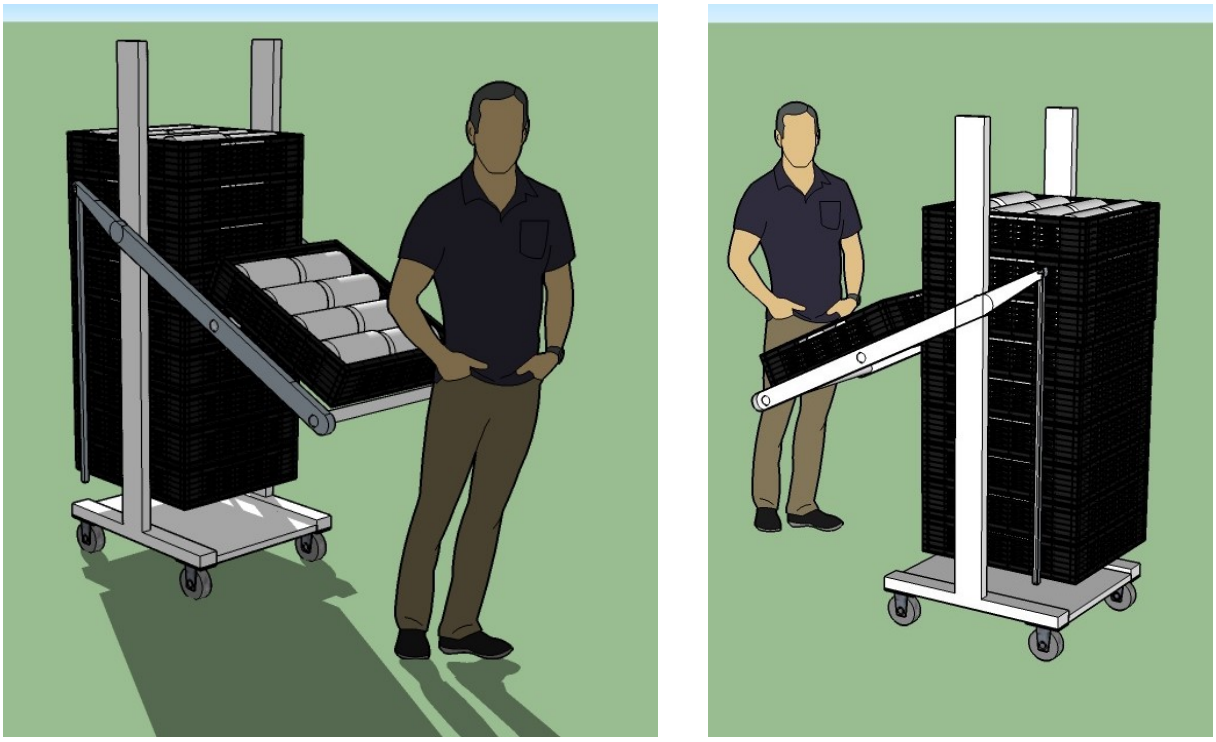


Figure 4.2: Two angles of the 3D model of a the conceptual trolley design.

The conceptual trolley has a mechanism that allows the crates to be unstacked and stacked simultaneously while creeling. The same mechanism allows creelers to load and unload crate stacks from the trolley. It can be seen in figure 4.2, that the trolley has a lever, which when in its full up position, locks onto the lowest crate. The lever will then be pushed down, lifting the entire stack, opening a slot at the bottom for a crate. The creeler will then pull the top crate off the stack and shift it into the loading position. The loading position is at hip level optimising it ergonomically for the creeler. The creeler can unload the bobbins and place them into the creel with ease and speed. Once the creeling of the bobbins is complete, the empty crate is placed at the bottom of the stack, and the lever is again lifted where the process is repeated until the all the crates are empty.

Thus far we still have not eliminated the need for labels, but the crates will improve the order that the bobbins arrive in. When the winders assemble the bobbins they do this in the order they need to be creeled. With the current bins, the order is lost when the bobbin falls into the bin. However, if only eight bobbins are in the crate, there are only eight positions. Looking at figure 4.3 the ID numbers of bobbin positions, it is evident that numbers 1095-1104 are not far apart. This will already improve the creeling rate, as less time is spent finding the correct bobbin position.

Figure 4.3: A picture showing the layout of bobbin position numbers in a creel.

If the crates had dividers in them to ensure that the bobbins could not be rearranged, the need for labels would be eliminated. If a crate with eight positions had dividers and it was known that the top right bobbin is, for example, bobbin position one and after that the positions followed a logical order, then there was no need for labels. For example, if the first stack of a colour arrived, it would already be packed in such a way that bobbins for positions one to eight would be in the top stack, with bobbins 72 to 80 in the bottom crate. That would mean the creeler would already know where the bobbins needed to go without the inspection of a label. The next stack of colour that would arrive would include bobbins 81 to 160.

As seen in the data of the number of bobbins per colour, there were not always bobbins that spanned across the entire 1104 positions. The bobbins were randomly distributed in the bobbin positions of a colour. If there were only 500 bobbins in a colour, every second bobbin position did not receive a bobbin. In these cases, the position was skipped in the crate, indicating the empty bobbin position.

If labels were removed from the process, it would not only benefit the creeling, but the winding as well. The doffing time, section 3.1.6, could be improved by 11 seconds, thus there would no longer be a label that needed to be applied. The breakdown probability of the winder would also be improved to a probability of 0.05, as most breakdowns were caused by the current integration of label application and the winder operation. The new breakdown probability was determined by doing a time study of winders that did not apply labels, but still wound the same material at the same speed. On average the machine completed 100 runs with only five breakdowns.

However, it needed to be determined if the winders would be able to handle the crates. There would need to be machinery designed and manufactured that allowed the winders to load and automatically stack and unstack crates. The technology had already been well developed and according to the head engineer of MONN, could easily be manufactured by their workshop.

The concept design (figure 4.4) allows the winders to have access to crates and gives them the ability to load a crate stack automatically. There are two chain conveyors where one provides the winder with a crate stack, and the other retrieves a full stack. The conveyor that feeds crate stacks to the winder will carry ten stacks. This is done to avoid the winders from having to wait for incoming crate stacks. The manner in which the crates stacks are taken from, or brought to the winder is determined by the material handling system.

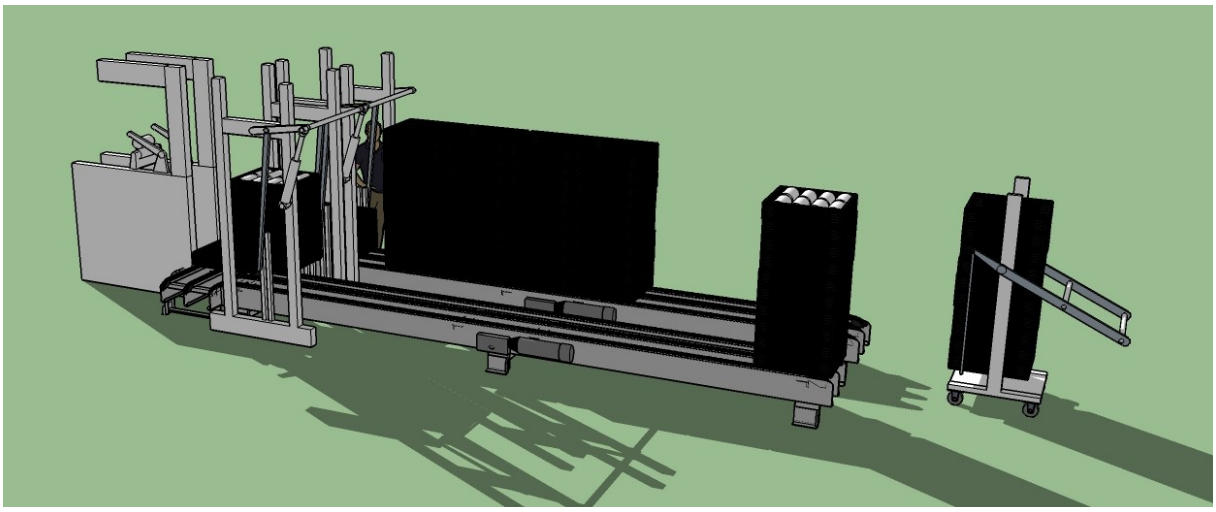


Figure 4.4: 3D conceptual design of a winder with an in-feed conveyor supplying it with empty crate stacks and another conveyor for the outbound full crate stacks. The automatic stacking and de-stacking machine units can also be seen (middle), with the winder feeding bobbins (left).

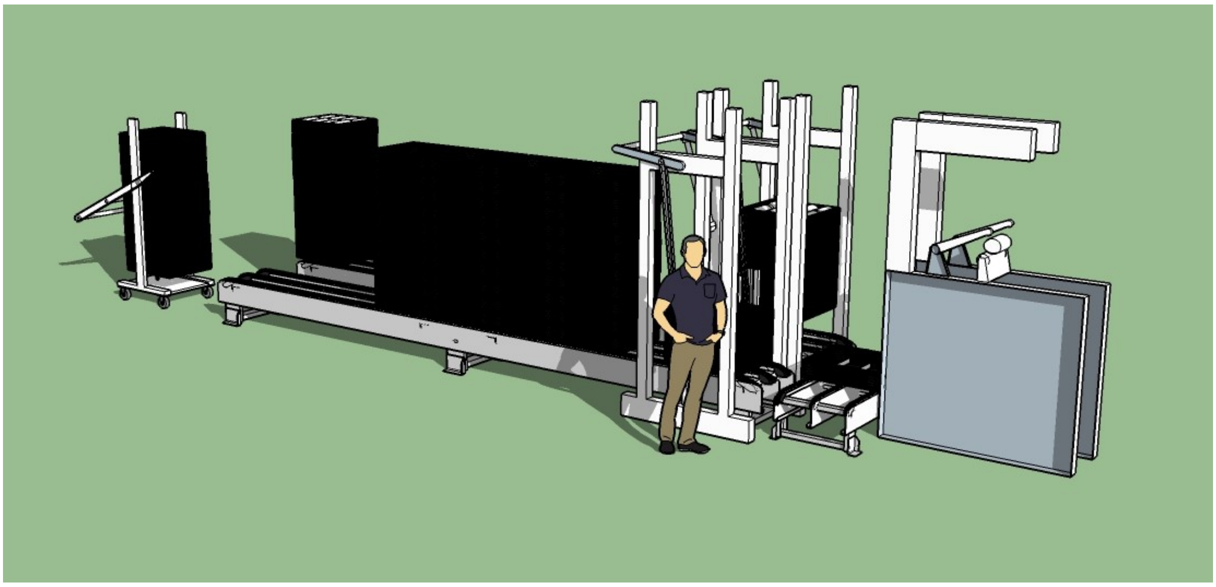


Figure 4.5: 3D conceptual design of a winder with a in-feed conveyor supplying it with empty crate stacks and another conveyor for the outbound full crate stacks. The automatic stacking and de-stacking machine units can be seen (middle) with the winder feeding bobbins (right).

If the crate system with no labels on the bobbins were used, there would need to be precise control of the crate stacks, from the winding through to the creels. It would be a challenging task as the number of stacks would greatly outnumber the number of bins currently used. The number of crate stacks had yet to be determined; the simulation models would also serve as a tool to determine the number of bins required. With the complexity of the bobbin handling increased, material handling systems would need to be developed to monitor and control the crates.

Initial trail simulation runs according to this creeling method, revealed a problem when trying to eliminate labels. There were cases where there could be 30 bobbins per colour scattered throughout the 1104 possible bobbin positions in a colour. If no labels were applied it meant that 14 crates would need to be sent to the creels. That would be due to the fact that the number of open positions between bobbins within the crates, determined the position the bobbin would be in the creel. For example, if there were a bobbin on position one, and the next bobbin needed to be put on position 100, there needed to be 98 empty slots in the crates to indicate the position. The high volume of empty crate stacks sent, were inefficient as they were using up transport time, space, and the crate stack could not be reused until the creeler had confirmed that there were no bobbins within the crate stack (Kulwiec, 1985).

Therefore labelling bobbins would remain in the system to avoid empty crate stacks. The labelling process would be modernised and, unlike the current system, this would happen when the bobbin was loaded into the crate after the winding of the bobbin had been completed. That would mean that the winders could start winding the next bobbin while the label was being applied. Breakdown probability of the winders would remain at 0.05 as it was assumed the new labelling machinery would not make mistakes. That was a valid assumption, according to the mechanical engineer of MONN, because the separation of labelling and winding meant the labelling application could be more controlled. The doffing time remained at 11 seconds as the winder had no input regarding the labelling process. As stated in 1.5, the two material handle system alternatives, needed to be developed to manage the crates.

4.1.1 Scenario 1

If material handling system warrants using mechanisation and computerisation it can improve efficiency and reliability (Kulwiec, 1985). However, it is important to note that one should not automate for the sake of automating, as it can be counter-productive.

The crate stacks needed to be handled with control and order, and it was a good idea to eliminate human input in the material handling process. People made mistakes and where control was essential, human error could waste time and become expensive.

When looking for a solution to a problem, it is essential that the surroundings and environment be understood as they could help you come to a solution. MONN recently sold off two old looms with their creels. That opened up a large area on the factory floor plan. The area opened was about 50m×30m with a platform above, of the same size. Like the looms used now, those platforms hosted the creels of the looms, and all platforms were at the same height.

The newly opened areas could provide a storage area for the crate stacks waiting for looms (figure 4.6). The current area measures 20m×20m meaning that if only the platform was used there would be a significant improvement in the storage area.

Currently MONN has space problems and it is undergoing projects to improve floor plan (figure 1.2) efficiency. This is important to note when designing the new material handling system, so that it will also use space efficiently.

If the winders could move from their current position to under the platform, it would open up space for machinery and other items that could not fit under the platform. If the platform could then be the crate stack storage area, it would be within close proximity to the winders below.

Figure 4.6: Area of the space available on the platform.

To keep the process automated there would need to be a hoist unit that would transfer full crate stacks to the platform and in the same manner, keep the winders populated with empty crate stacks. The platform would need equipment that would move the stacks to and from the hoist unit, as well as manage crate stacks needing to go to the looms.

The movement of the crate stacks had to be minimised and kept simple (Kulwiec, 1985). If crates were to enter from the one side of the platform and leave for the looms on the opposite side, it would make controlling crates more straightforward. That might well be done by a gravity system. However, the distance they had to cover would require a significant height difference between the two sides. It would be more controlled, as well as more cost effective if MONN were to manufacture their own chain conveyors to move the crate stacks. The platform area above allowed for a total of 26 conveyors with a length of 30 meters, given the crate dimensions that translated into 30 crates per chain conveyor.

The transportation of the crate stacks to and from the looms was done by an overhead mono-rail guided unit, supplied by D-MAC. That eliminated the need for a person and a forklift to transport the crate stacks. The creels would receive the crate stacks, and the creelers would collect the stack with the trolley.

That would be more expensive and a more modern system. Its performance would be determined by modelling the new design. It would reduce the amount of human input required, which meant two fewer people needed.

It is MONN's decision as to what will happen to the employees whose jobs have become redundant but an additional iteration was run where all these employees will be moved to creeling, and the results will be documented as a separate scenario, ***Scenario 1b****.*

4.1.2 Scenario 2

The objective of the second scenario was to use the same crates and creeling method as in scenario 1, but with a material handling system that would not be too expensive and would require minimal change to the current system.

The WinderMachine modules would undergo the modifications as done in the previous scenario they would, however, remain in the same location as they were currently. The current bin storage area would become the crate stack storage system. The storage area would need clear demarcation and good control, so that crate stacks were not confused.

When a crate stack had been completed by the winder, the employee would collect the stack from the outgoing conveyor with a trolley (see figure 4.2 as used in the creels). The crate stack would then be placed in the crate storage area.

The transport of crates to and from the loom would be completed by the same forklift as used currently. That design would have a higher risk of failure, where crates stacks had been confused, as there was far more human input involved. However, it required less investment than scenario 1 and 1b.

4.2 Programming the scenario models

The models programmed for the two scenarios used the model of the current system. Below would be discussed what needed to be altered, and how the model was adapted for the new scenarios.

4.2.1 Scenario 1

Winding Module

The first changes seen in the Winding module were, where the modules, WinderMachines, needed to be kept under the platform where the old looms used to stand. On top of this they also needed to be modified so that they could fill crates instead of bins. Receiving and sending conveyors were added to the machines. Figure 4.7 shows the process, blocks and conveyor units put in place to allow for this. The logic of trying to maintain ten crate stacks in the incoming convey is managed by state charts (figure 4.8).

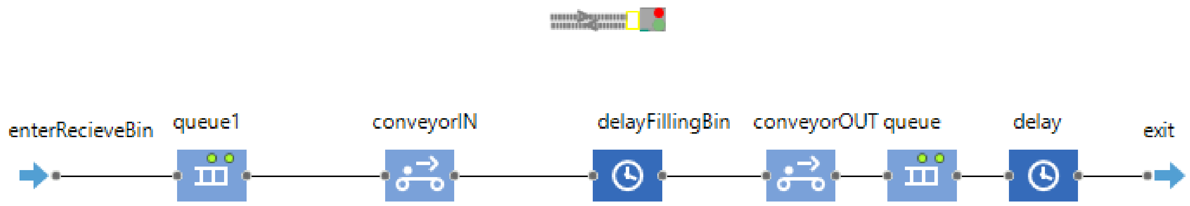


Figure 4.7: Winder with additional process blocks and conveyor units, updated WinderMachine icon can be seen above.

The state chart (figure 3.14) managing the state of the winder needs to be adjusted as the winder no longer needs to stop when a stack of crates is full. That is because the automated loading system always ensure a crate is in place for loading. The breakdown parameter of the winder needs adjusting at it is now 0.05 instead of 0.1, as a result of the labelling application alteration.

The bin agent was modified to suit the dimensions of the crates. The winder machine logic needed to be altered as each stack carried 80 bobbins, unlike the original bins where it was dependant on the size of the bobbins.

The same communication between winder machine and winding modules as in the current base model and all breakdowns and start-up, would still be managed by an operator unit. However, the resource unit for collecting and bringing crate stacks was a hoist unit instead of an operator unit.

The hoist unit was introduced into the winding module with all its destination point mapped out accordingly. The hoist can travel as 2 m/s, and the lifting/dropping of the crate stacks is done between the starting and stopping destination points. Owing to the reduction in work, one operator unit fewer was required in the winding module.

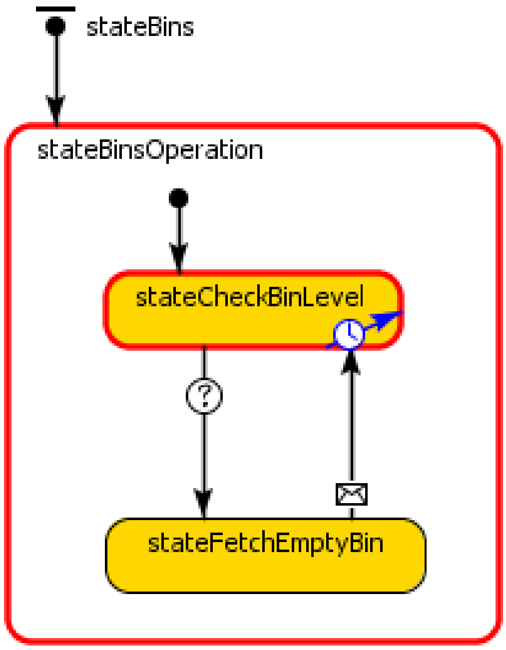


Figure 4.8: State chart within the winder managing the number of empty crate stacks on the incoming conveyors.

Crate Storage

Individual CONVEYOR (figure 4.9) modules, were introduced into a module called CRATESTORAGE (figure 4.10), which would make up the storage system of the crate stacks. Firstly, a conveyor needed to have the ability to make empty crate stacks received from loom, available to winders, as well as make full crate stacks available to the looms. Each Conveyor module was equipped with two conveyors, but only one could be used at a time, either crate stacks to or from the loom.

When a crate stack is sent to the crate storage from the winding, the model first determines if there is space to satisfy the request. The model does this by checking the number of crate stacks on each conveyor belt and the number of crate stacks inbound to the conveyor in question. If a conveyor belt is empty the crate is sent, and the belt is immediately assigned to the job ID given to the crate stack; but if there are conveyor belts with matching varJobIDs and with space available the crate stack is sent there before using an empty conveyor belt.

A similar approach is used when empty crate stacks are sent from the looms to the crate storage. The model will first look for varJobID of the conveyor belts equal to 9999, the code for returning crates stacks. To avoid a situation where winders cannot access empty crates due to all conveyors being full, one of the 26 conveyor belts is reserved for empty crate stacks.

The changing of the method of transportation from forklift to mono rail D-MAC unit was a matter of creating handover points, a network of rails, and changing the resource unit to the mono rail unit. The mono rail unit travelled at 2 m/s.

Loom Module

The new creeling method does not need much change in the loom module. The major change is that creelers need to acquire a trolley before moving the crate stacks to the creeling position. This can be done by altering the network topology and adding a new sequence of process blocks as illustrated by figure 4.11. It includes modifying the process of sending empty crate stacks

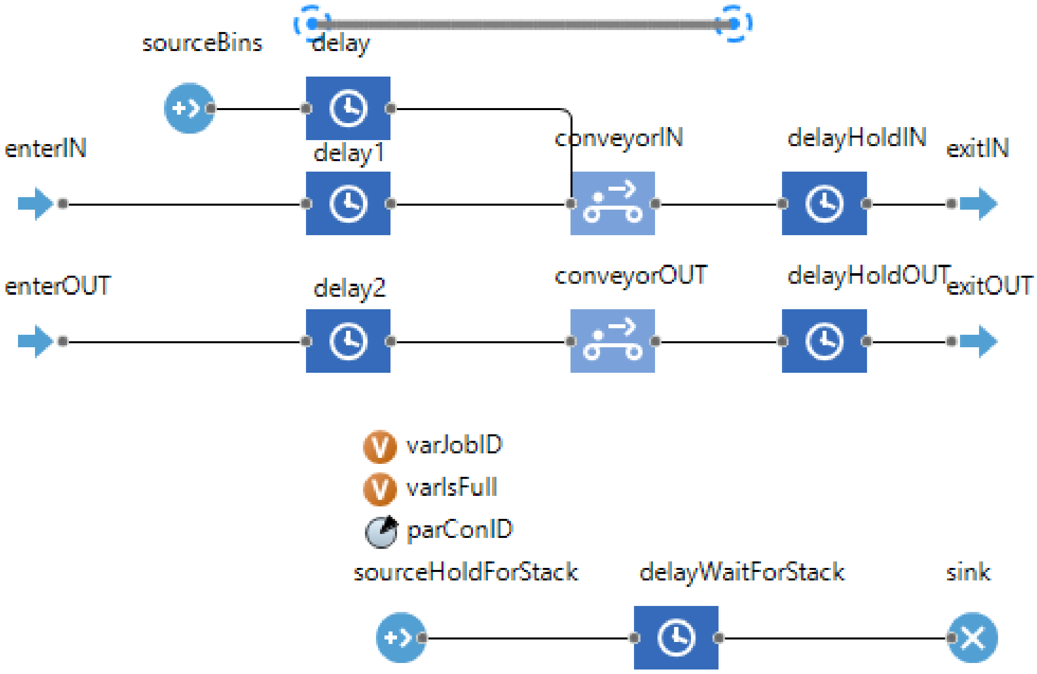


Figure 4.9: The StorageConveyor module currently hosting empty crates.

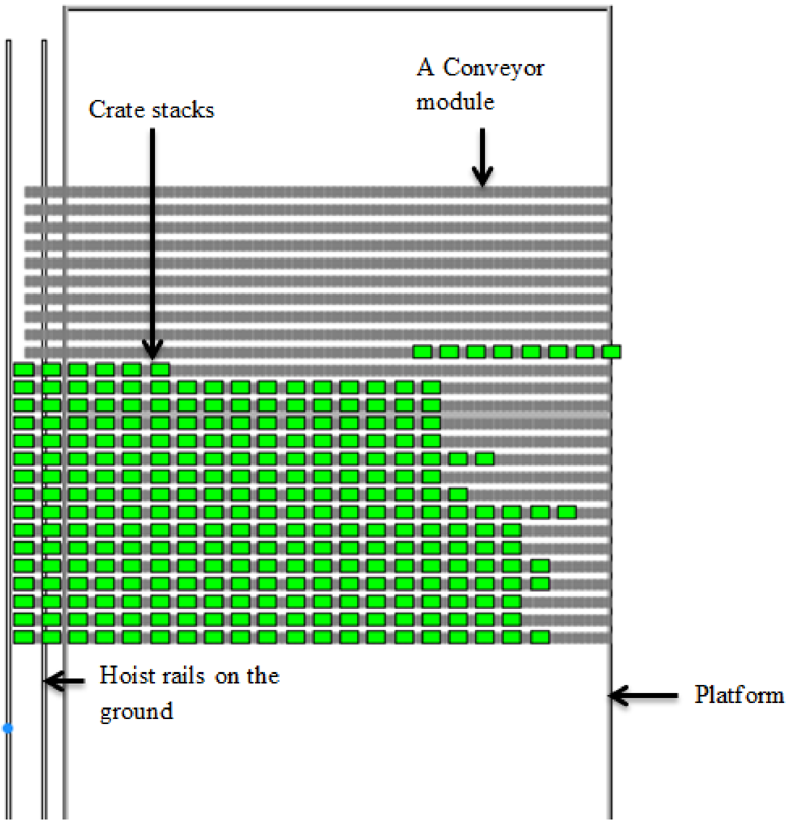


Figure 4.10: 26 conveyor modules forming the CrateStorage module.

back to crate storage.

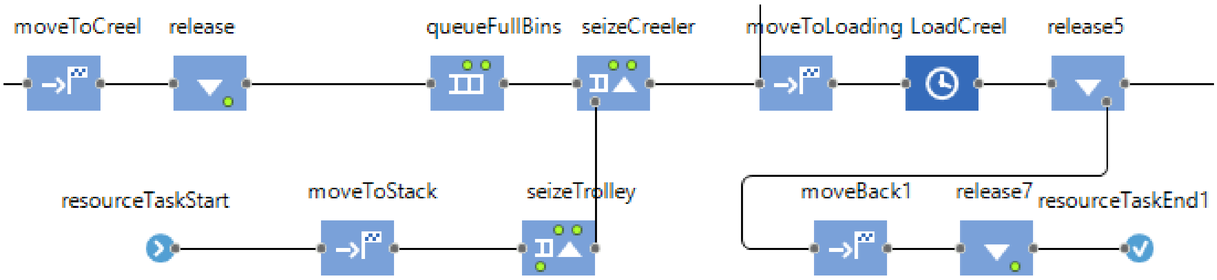


Figure 4.11: The process blocks allowing creelers to fetch trolleys to collect crate stacks.

A time study was done where a mock-up of the new creeling system was run for over 80 bobbins, a full stack. A basic mock-up of the trolley was built to try and accurately depict what it would be like using a real trolley. The creeling method was explained to three creelers, and three trials with three creelers were completed. Even though the bobbins still had labels, being packed in order, was beneficial and the average result of the three trial runs was 80 bobbins in 17 minutes. This translated to a creeling rate of four hours per 1104 bobbins. The creeling rate was adjusted accordingly.

To determine the optimum value for the parameter, the number of jobs held in crate storage area was determined by running optimisation experiments. It was done to determine whether the changeover time and total time of the job in the system were the most favourable values. The result was input where the number of jobs held in crate storage area, was six.

This concluded scenario 1's programming adjustments.

Layout of Scenario 1

Figure 4.12 shows a screen shot of scenario 1 model layout, while running an iteration. It is clear that the winders are all under the platform. All winders have the ability to send or receive crate stacks from the hoist unit. All winders are processing jobs and crate stacks are being filled. The crate storage system can be seen functioning as there are full crate stacks awaiting a call from the looms. There are also empty crate stacks waiting for a call from winders. The hoist unit is awaiting instructions to either collect or move crate stacks. The mono rail unit is currently standing idle as all looms have their demands met at that time. There are three looms in production indicated by green dots and one loom in set-up state with yellow dots. One loom is standing without work indicated by a red dot. The trolleys to handle these crate stacks can be seen parked as there is no creeling work at this time.

4.2.2 Scenario 1b

For scenario 1b the model is exactly the same as with scenario 1 but workers have been relocated from previous positions to the creeling. One worker was moved from the winding as they were no longer needed there due to the improvements in the winding section. The other worker was moved from forklift driving as the mono rail unit made their job of transporting crate stacks redundant.

4.2.3 Scenario 2

From a programming perspective scenario 2 was a combination of both the original base model and scenario 1. The Winders needed to be equipped with the same conveyor system as the winding process in scenario 1. However, the winders did not need to move to the platform, but rather stay put. The state charts managing the winders and incoming crates were identical to scenario 1 (figure 4.4).

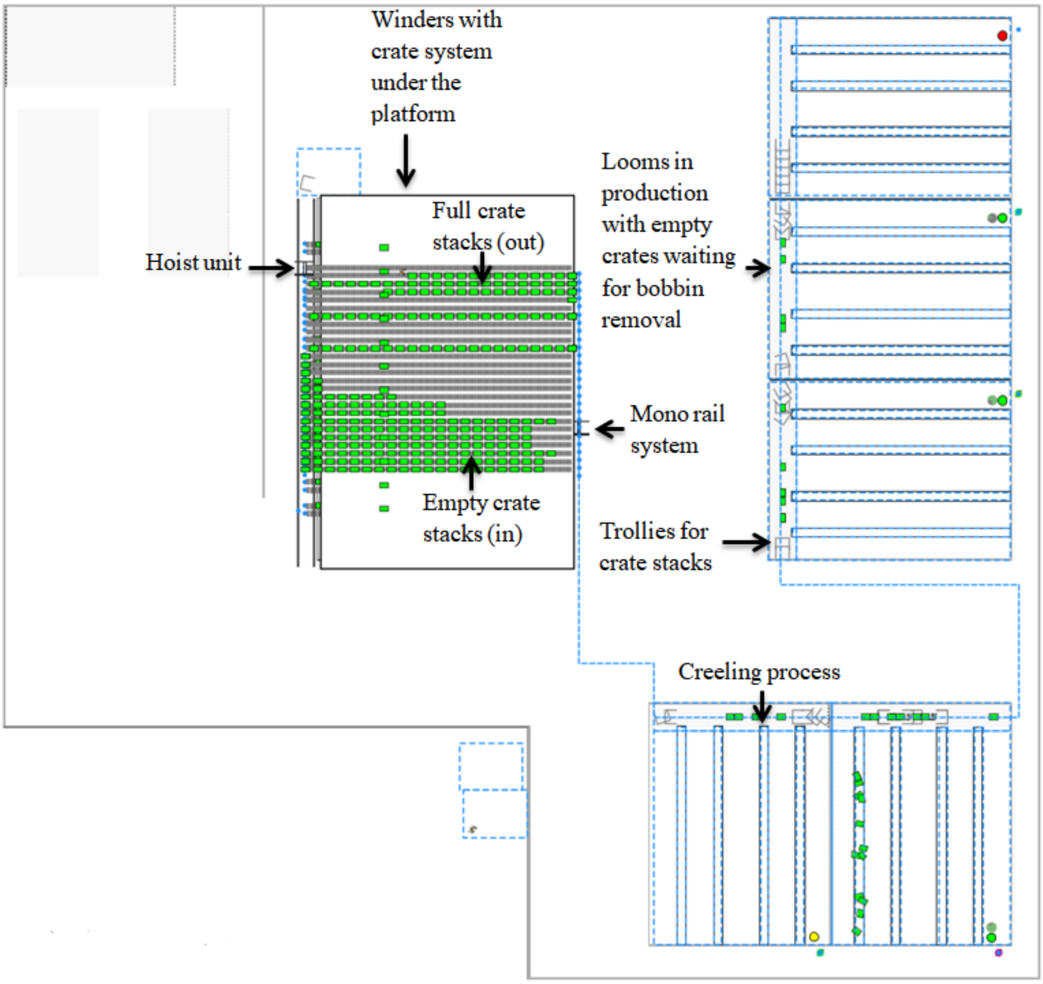


Figure 4.12: A screen shot of scenario 1 model layout while running a iteration.

The retrieval of full crate stacks or requests for empty crate stacks were done by resource unit worker, but as with creeling, they needed first to collect a trolley to make moving crate stacks to the storage area easier. That meant that as the original base model resource unit worker was responsible for all winders' crate stack requests as well as startup and breakdowns. The work load would have needed two operators in the WINDING module.

The transportation of full and empty crate stacks had the same configuration as the original base model where a forklift was used. The LOOM module also responsible for creeling, was identical to that of scenario 2. However, the requests for crate stack transportation was for a resource unit forklift.

Layout of Scenario 2

Figure 4.13 shows a screen shot of scenario 2 model layout while running an iteration. All winders have the ability to send or receive crate stacks from the worker who uses a trolley to move the crate stacks around. All winders are processing jobs and crate stacks are being filled, but a number are calling for a worker to perform a function as they indicate a yellow dot. There are also four winders with red dots indicating they are standing idle. There are a number of full crate stacks in the storage area waiting to be called by a loom. There are two looms in production indicated by green dots and two looms in set-up state with yellow dots. One loom is standing idle as there is no creeling work at this time; they can also be seen at the winding.

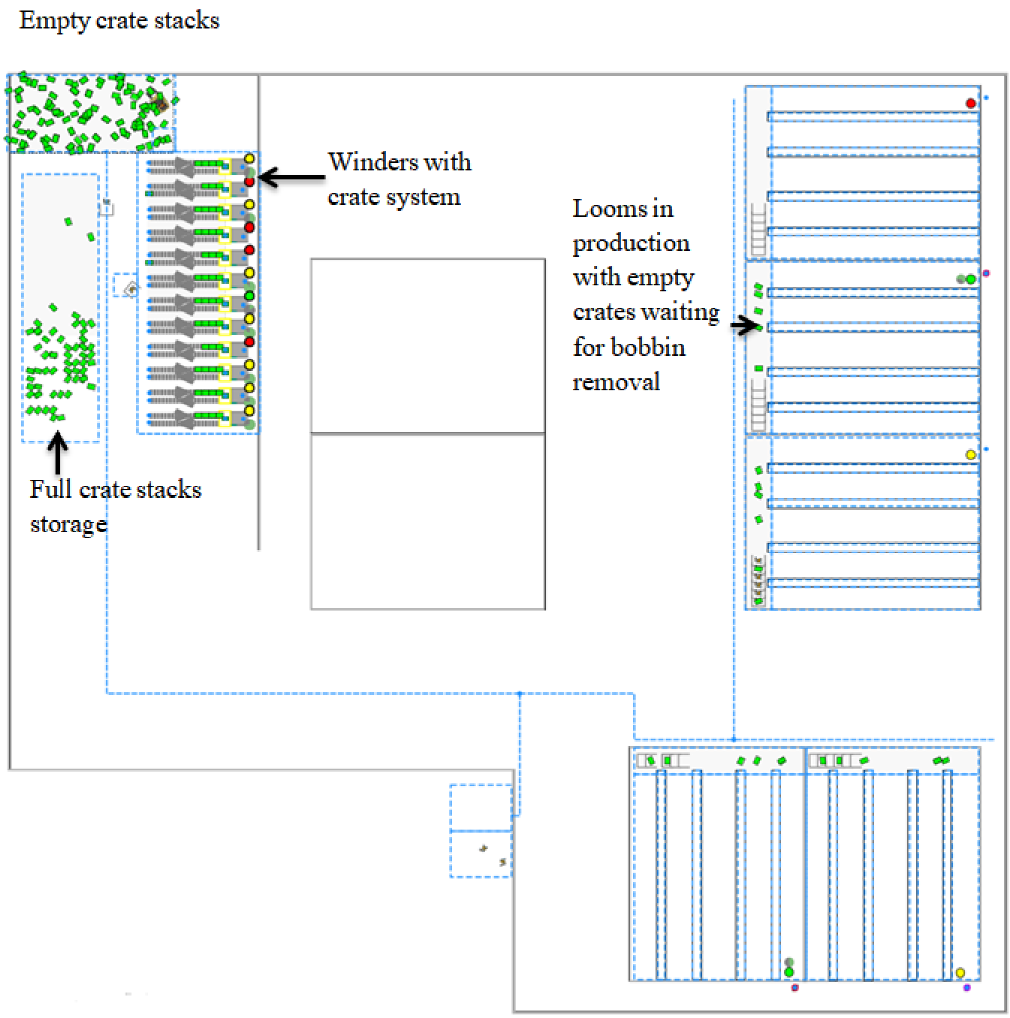


Figure 4.13: A screen shot of scenario 2 model layout while running a iteration.

4.2.4 Experiment for results

To determine the performance of the two scenarios the same experiment that was used for the base model could be utilised, as discussed in section 3.6.

4.3 Scenario Budget

4.3.1 Scenario 1

To budget for a project of this magnitude is challenging as many factors and variables need to be taken into account. MONN's engineer is confident that MONN would be able to design and build what is required. However, items such as gearboxes, motors, PLCs, and drives would be purchased. If MONN could build their own machinery such as the conveyors and crate stacking units, it would result in great savings as MONN was not paying someone who needed to make a profit. After much research was done by MONN, an initial budget of R7 million was established.

Companies like Bosch could be subcontracted to do the same work, but it would lead to the cost of anywhere from R15–R20 million rand.

4.3.2 Scenario 2

This scenario required less investment as there was no need for conveyors or the mono rail transport unit. However, it still required the modification done to the winders, which was a major part of the project. The budget from scenario one could be adjusted to the needs of scenario 2. The figure could be reduced to approximately R4 million.

Chapter 5

Results and discussion

The following results are a comparison of the results from experiments done with all the model scenarios, including the current base model. As mentioned in section 3.6, 100 instances would be run for each scenario, each instance having a unique random seed. However, the same set of 100 random seeds will be used for each of the scenarios. Each iteration was run for a virtual period of one year. Figure 5.1 shows four box plots of the mean change over times achieved by the different models. It can be seen that all new scenarios were an improvement on the current base model. It also shows that scenario 1b, which is the same as scenario 1 but with the excess workers relocated as discussed in section 4.1.1, gave the most favourable mean changeover time with more compact distributions.

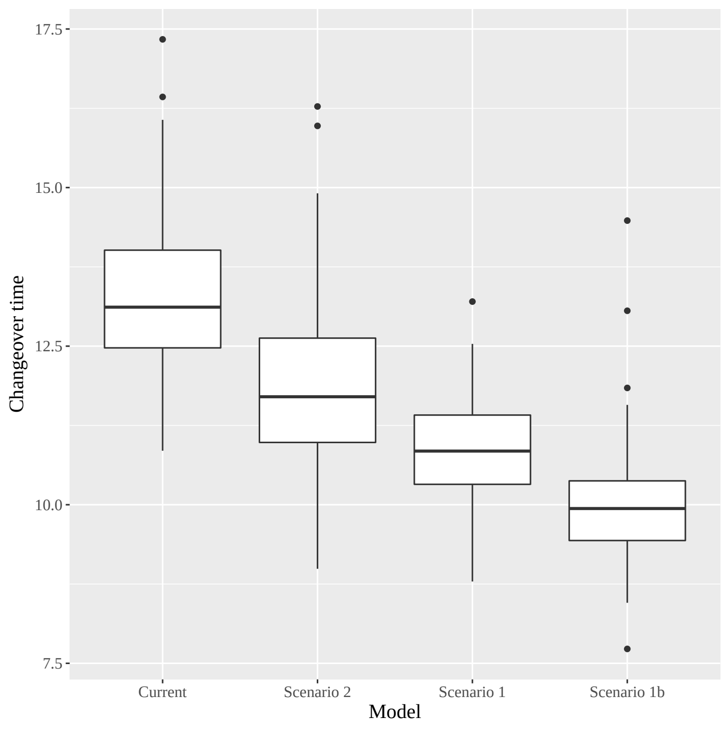


Figure 5.1: The box-plots representing the resultant data of mean changeover time for each model.

Section 1.5 mentioned if a scenario showed there was an improvement in the changeover, but not an improvement in the job time, then implementing that scenario was counter-productive. Figure 5.2 shows four box plots of the mean job time times achieved by the models. It can then be confirmed that all new scenarios are an improvement on the current base model. Just as with mean changeover time, it is clear that scenario 1b presented the most favourable mean job time.

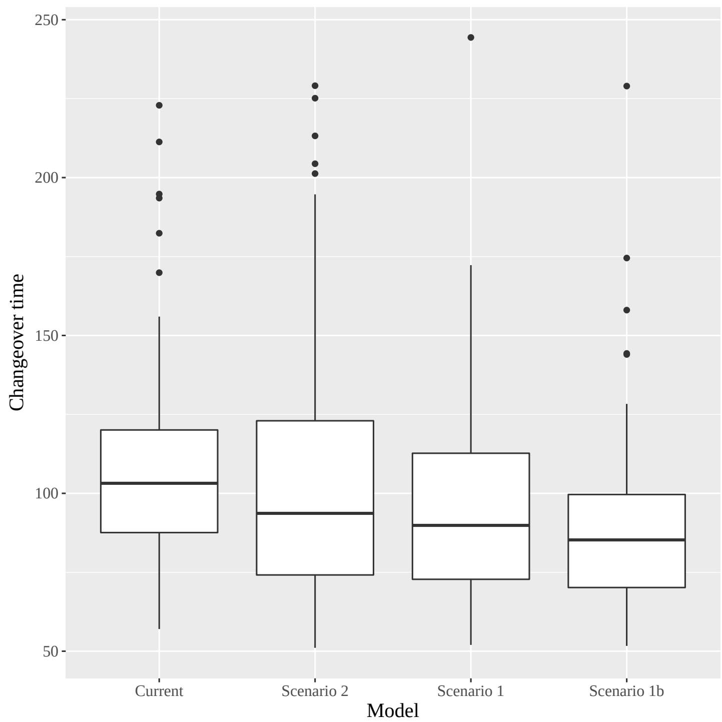


Figure 5.2: The box-plots representing the resultant data of mean changeover time for each model.

The results provided by scenario 1b indicated that the looms were in fact the bottleneck in the process. Owing to the addition of two creelers the overall rate of creeling was increased, and thus decreased the changeover time. The job time showed a significant benefit as well. That the loom created the bottleneck was consistent throughout all the models and could be seen when the models were run.

The following tables, 5.1 and 5.2, show the results achieved regarding mean changeover and mean job time as well as the intervals of these means with 95% certainty. For example, it was established that the base model could, with 95% confidence, produce a result where the mean change overtime and mean job time would fall between 12.99–13.46 h and 102.33–114.79 h respectively.

Table 5.1: Table showing results of the mean changover times and changeover times with 95 % confidance range, achieved by the models in h.

Model	**Lower bound**	**Mean**	**Upper bound**
Current	12.99	13.22	13.46
Scenario 1	10.71	10.87	11.03
Scenario 1b	9.79	9.97	10.16
Scenario 2	11.58	11.85	12.12

Scenario 2 has wider distributions in both mean changeover time and mean job time when compared to the other models. This phenomenon was seen in both the box plots and the confidence intervals. That indicated that scenario 2 was sensitive to its inputs, which implied that new systems in place had a higher risk of producing unpredictable changeover times.

To ensure that the resultant distributions from the new scenarios were independent of the current base model's resultant distributions a chi-square test was conducted. In both cases of

Table 5.2: Table showing results of the mean job times and job times with 95 % confidance range, achieved by the models in h.

Model	Lower bound	Mean	Upper bound
Current	102.32	108.56	114.79
Scenario 1	89.76	95.56	101.35
Scenario 1b	83.59	88.85	94.11
Scenario 2	95.90	103.73	111.56

comparing scenario 1, 1b and 2 to the current base model, the p value achieved were greater than the significance level of 0.05, which concluded that the data sets were independent of one another.

5.1 Sensitivity analysis

To model the two new scenarios, two major assumptions were made. One of which was that there was never any error in handling the crate stacks between the winding and the creels. The other assumption made was that there were no errors or breakdowns that happened when a bobbin was labelled.

When the bobbins were labelled there might be an unexpected error that could occur like a bobbin being out of position. In that event, the winder operator was given two minutes to fix the problem. That was considered to be a comfortable amount of time as the labelling error would not be regarded as a major breakdown.

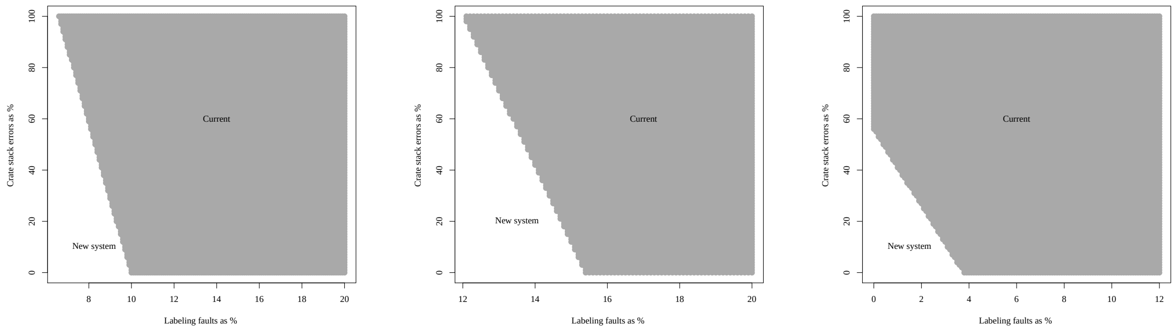
Owing to the layout of the arriving crate stacks of scenario 1 and 1b, the creelers might confuse two crate stacks at the creel and would need to identify the problem to rectify it. It was estimated that the problem would be corrected in five minutes as it was just a case of swapping two crate stacks around. In the case of scenario 2 a foreseeable error in the crate handling was that the forklift might pick up an incorrect crate stack, or that the winder operators confused the stacks. If that error would occur, an estimate of ten minutes was given to resolve problem, as the crates stacks in question only needed to be identified and the issue resolved.

Multi-dimensional sensitivity analysis was completed to determine how many errors could happen before the new scenario was no longer an improvement. Before that could be done, more data had to be gathered from the models. It included the mean number of crate stacks per job and the mean number of bobbins assembled, per job. Owing to both models using the same seed values, the results were identical with mean numbers of 3840 bobbins per job and 52 crates per job.

The mean job times were compared as they were a representation of the performance of the entire system. To remain conservative, it was assumed that if an error did occur the whole job would go on hold for the duration it took to rectify the error. For example, the mean job time for scenario 1 was 95.56 h and when comparing that to the mean job time of the current base model the difference was 13 h in favour of scenario 1. That implied that MONN could afford a total of 13 h of errors. For example, if we were looking at just label errors a maximum of 390 label errors could occur before the new scenario had the same mean job time as the current base model. However, it could be the case that both errors would occur, and so the figures 5.3a, 5.3b and 5.3c were developed.

The transition from light to the dark area is the where the scenario in question is no longer an improvement. The shaded area represents an area where the current system will have a better mean job time.

From figure 5.3a it can be derived that the model for scenario 1 is not as sensitive to crate



(a) Scenario 1 (b) Scenario 1b (c) Scenario 2

Figure 5.3: Sensitivity analysis

errors as it is to labelling errors. This is because there is a large volume of labels per job and not as many crate stacks. If MONN can manage 20% errors in crate stacks and 7% in labelling in both scenarios, they will still see a substantial improvement in performance. However, anything lower will always be beneficial. A very realistic achievable target for MONN will be 5% crate errors and 3% labelling errors, and if achieved, MONN will benefit largely. Figure 5.3b shows the scenario 1b where MONN has even more tolerance.

Scenario 2 (figure 5.3c) has similar properties. However, this system can only afford 4.82 *h* of errors, and a crate stack error needs ten minutes to be rectified. This means MONN needs to aim to have total labelling errors no higher than 2% and crate stack errors no greater than 10% to still see a notable improvement.

5.2 Discussion

Although all new scenarios show improvement, the cost of the project needs to be taken into account to be justifiable. When running the simulation for all models, it is clear that when there are many jobs being processed, the looms form a bottleneck and thus jobs queue up in front of the looms. This is because the winders are producing the bobbins faster than the looms are using them. Therefore, all calculations done, assume that the looms will always have a job waiting for them. This may sound unrealistic, but it is a conservative approach when working out whether the new scenarios are justifiable. If this happens, the factory capacity is then determined by the looms. Therefore, any addition to the weaving time will increase the capacity of the factory. An improvement in job time does not necessarily indicate the amount of additional weaving capacity MONN will gain. An improvement in changeover time will result in the looms being able to start weaving sooner and so increase the capacity as the loom stands idle for a shorter period of time.

A loom spends a mean time 35.77 *h* weaving on a job with the mean size of 800 square meters. If scenario 1 is implemented the mean changeover time improves by 2.353 *h*. Thus the loom stands for 2.353 *h* fewer between jobs. With the hourly cost of a loom currently at R920 an hour, this results in a saving of R2164.76 per job and R408158.00 annually based on 190 jobs. Most importantly the loom can now weave for an additional 2.353 *h*. By taking 2.353 and dividing this by mean weaving time of 35.77 *h* it shows the loom has an additional 6.59% in weaving time, and so 6.59% increase in the loom's capacity. Due to MONN having five looms there would be an increase of 32.96% (table 5.3) of one looms capacity to the total capacity of the factory.

Today a new loom costs 1.25 million euro which translates into R19.57 million. If a hypothetical company had one loom, and purchased another loom — assuming the loom always had work— the capacity of the factory will increase by 100%. According to MONN's directors, a loom is depreciated over 10 years meaning the loom is payed off in 10 years, it contributes R1.957 million per annum in paying itself off. Meaning a portion of the companies revenue is deducted to pay for the loom. The same can apply for the the new scenarios as the additional capacity provided can help pay-off the project. Therefore 1 % capacity contributes an additional R19 750 per annum in paying itself off. The increase of 32.96% of one looms capacity can contribute R638 746.86 per annum to project pay-off amount, provided there be work for this additional capacity.

In addition to this scenario 1 does offer other cost-effective savings which can contribute to its annual payback amount, for instance, two employees fewer will be required in the process and the reduction of loom idle time. This totals R408 158.00 savings because of the reduction of loom idle time and the R142 090 saved because of employee reduction. This amounts to R550 300.40 per annum. To this add an equipment maintenance fee of R100 000 a year, it can be estimated a net pay-off of R1 088 994.85 per annum.

By knowing the pay-off period, MONN can get a good idea whether the scenario will be feasible. It must be noted that the additional capacity will also increase profits. These figures are confidential, which will result in a more conservative approach.

Loom Price: R19 570 000

Depreciation period: 10 years

Contribution to pay-off per annum: R1 957 000

Contribution to payoff per Annum per % capacity: R19 570

Mean Loom weaving Time: 35.77 h

Standing Loom costs : R920

Table 5.3: Table showing the annual amount that a scenario can contribute to its pay-off.

	Unit of measure	**Scenario 1**	**Scenario 1b**	**Scenario 2**
Change over difference	h	2.34	3.25	1.38
Total capacity increase	% of total	6.59	9.112	3.850
Loom effective capacity increase	% of one loom	32.639	45.46	19.248
Pay-off earned annually	R	638 746.86	889 595.19	376 682.84
Changeover Savings	R	408 158.00	568 449.60	240 699.60
Staff reduction savings	R	142 090	N/A	N/A
Maintenance costs	R	(100 000)	(100 000)	(100 000)
Total pay-off annually	R	**1 088 994.85**	**1 358 044.79**	**517 382.43**

A similar approach has been favoured for all other scenarios. Table 5.3 summarises the findings. For scenario 1b a significant change was that the employees were transferred to the creeling instead of being let go. The table showed that scenario 1b would provide the largest amount of money annually in paying the project off. However, scenario 2 would be a cheaper investment.

Owing to the uncertainty regarding the exact amount that will be invested in each project, a graph 5.4, depicts the required number of years it takes to clear the investment: a scenario vs investment amount, per scenario.

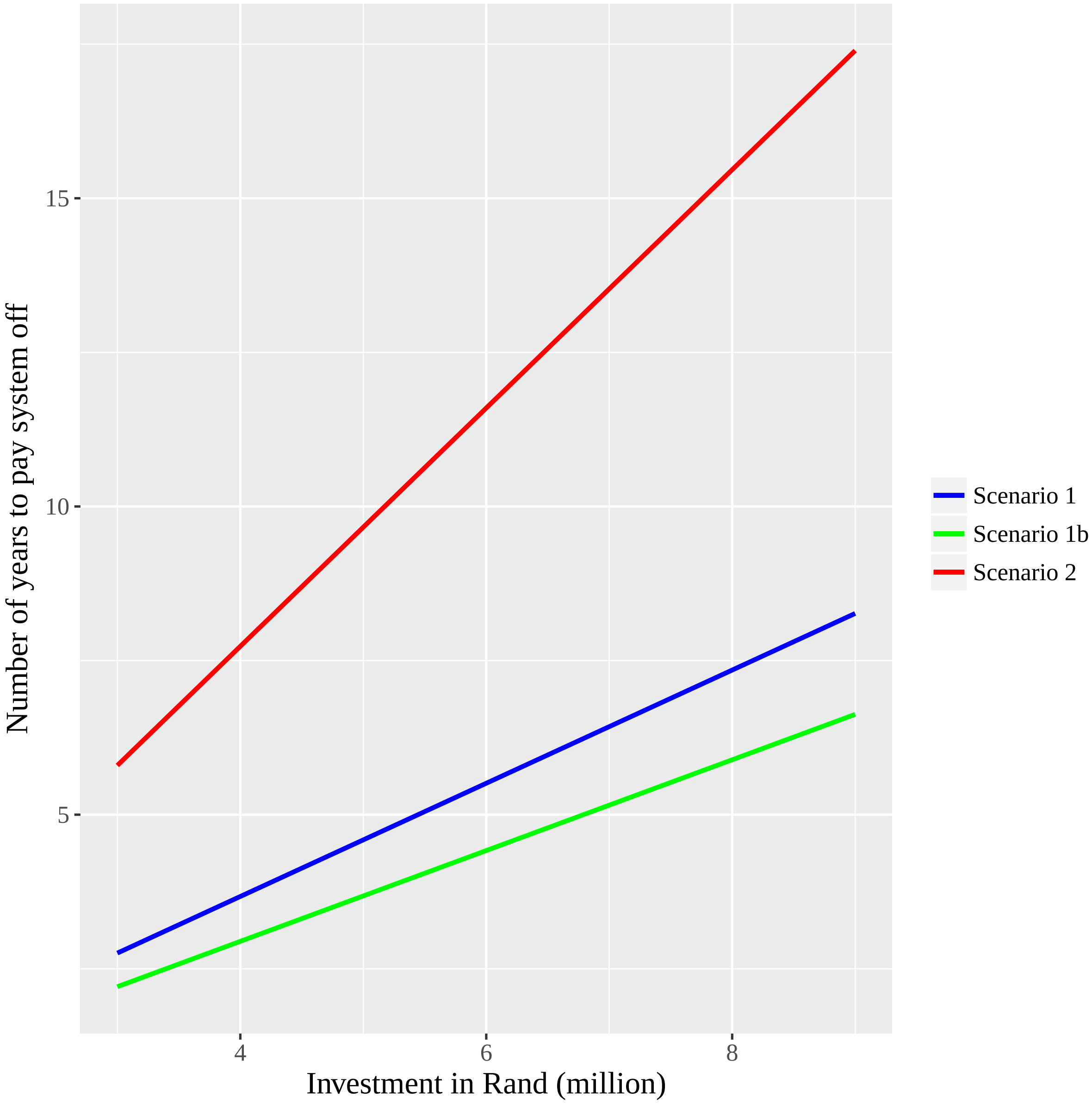


Figure 5.4: Graph showing the relation ship between the investment cost and number of year needed to pay itself off.

Chapter 6

Conclusion and Recommendation

All new scenarios show that — provided the projects do not go far over anticipated budget — they will pay themselves off in less than ten years, which means that the additional capacity that it provides will be cheaper than that of the capacity provided by a loom, making them very attractive in terms of value gained.

The results in graph 5.4 and table 5.3 show that scenario 1b will provide MONN with the best results with an anticipated pay-off period of just over five years, but other scenarios can be considered as they are feasible as well. Scenario 2 will pay itself off in 7.5 years, but any increase in budget will result in the figure being more than ten years. Scenario 1, much like 1b, is also a possible solution, but completely unnecessary to retrench employees. This will result in them not having work and will slow down the process so much, that it turns out to be more expensive. Depending on what MONN is prepared to spend will decide which scenario they must go for. If capital is an issue it will be an option to implement scenario 2 and then later upgrade it to scenario 1b.

The new material handling and creeling system can only reach its full potential provided the factory remains busy enough. If MONN does not foresee an increase in demand in the coming years, careful consideration must be made when implementing the project. If demand is said to increase it will not only save MONN money, but also improve customer satisfaction due to orders being processed faster. As a result of the savings made, MONN can decrease their product prices to increase the demand for their product.

At the implementation of any scenario, MONN must first convert one loom and four winders and get the processes running smoothly. If there is a flaw or problem it can be rectified. Once the system is running successfully for one loom the rest of the machines can be converted to the new scenario. This method of implementation is recommend to avoid any problems which will cause the factory to stand idle.

6.1 Recommendation for project improvement

In order to improve this project, data with larger sample sets must be acquired. This will lead to more accurate distributions that will be used in the simulations.

Another possibility will be to collect job data for the period of one year. The exact *job specifications* of each job will run through the factory making the all data distributions pertaining to job specification redundant. This could improve the realism of the jobs processed. However, jobs created by the current method are not unrealistic per se.

Although the models developed are already detailed, there is always room for improvement in that regard. One can look into the behaviour of the employees and how it affects their work output. The elimination of all assumptions will also provide more accurate models, but with the appropriate sensitivity analysis this is not necessary.

QUOD ERAT DEMONSTRANDUM

Bibliography

AnyLogic (2017). Anylogic simulation.

Biller, B. and Nelson, B. L. (2002). Answers to the top ten input modeling questions. *Proceedings of the 2002 Winter Simulation Conference*, pages 35–40.

Goubergen, D. V. and Landeghem, H. V. (2002). Rules for integrating fast changeover capabilities into new equipment design. *Robotics and Computer-Integrated Manufacturing*, 18(34):205 – 214. 11th International Conference on Flexible Automation and Intelligent Manufacturing.

Hammer, M. (1990). Reengineering work: Don't automate, obliterate. *Harvard Business review*, pages 104–112.

Kulwiec, R. A. (1985). *Materials Handling Handbook*. John Wiley & Sons, second edition.

Law, A. M. and McComas, M. G. (2001). How to build valid and credible simulation models. *Proceedings of the 2001 Winter Simulation Conference*, pages 23–29.

Luttmann, A. and Jajer, M. (1992). Electromyographical field studies in a weaving mill. *Redustion Of Muscular Strain by Work Design*, pages 593–602.

Manson, N. (2006). Is operations research really research. *ORiON*, 22(2):155–180.

Mileham, R. M. S. C. G. G. T. and Owen, G. (1996). An assessment of the role of design in the improvement of changeover performance. *International Journal of Operations & Production Management*, 16(9):5 – 22.

Tabibzadeh, K. (1989). Simulation system for material handling system design. *Computers ind. Engng*, 17:270–273.

Appendix A

Reflection on learning

From a young age I was actively, with a rather strong hand from my father, involved at the factory. I started at the bottom of the ranks, at the age of seven at the sampling department, to working my way up to the workshop in Grade 8, to finally a more senior role of conducting the odd projects. That being said, I have always had an interest in the business and working there is something I am looking forward to.

The second year at university was the first time I was exposed to a class offered by the industrial engineering department, BPJ 210. The concepts discussed in that class briefly scratched the surface of industrial engineering in its entirety, but it was enough to give me a taste of what it was about. I could not help but start thinking about things in our factory relevant to industrial engineering. I slowly started applying what I was learning as I went along.

In the June holidays of my third year, I worked at the factory for three weeks or so and I discovered a problem. The problem you would know as the loom changeover time problem discussed in this project. I approached my father about it and mentioned the creeling system not being up to standard; doing some quick calculations together he soon agreed something needed to be done. If a project like this is successfully implemented, it will make the company more competitive, allowing it to grow and create more jobs and assist in developing the local economy.

Starting the second semester the simulation project deliverable was made available, and my group and I chose to use this problem as bases for the project. What materialised was a model that was far too complicated and could not even open on some computers, due to the rather ridiculous number of state charts and variables.

This project still shows a lot of potential and deserves to be a final year project. The final year project is not what I expected. Over the years of being at university it has escalated to be the pinnacle of my studies; this virtual wall I need to climb before I graduate. Yet it does not feel like a big wall. It is so far, a somewhat pleasant experience, and it seems like I am working on something meaningful rather than a rushed assignment I just want to complete.

Like certain relationships with people, there comes a stage where you have spent enough time together for a while. I definitely get that feeling about this project when doing the literature review and, more painfully, the research that comes with it. There is nothing more frustrating than reading a good and relevant abstract and being denied access to it. This is where I learned that UP library is your friend and provides many related resources; you just need to find them. Finding them becomes an art that I began to master too late; unfortunately after countless h have been engraved into the project.

Something I have learned is that the monster that I may think that Latex is, is a rather simple and easy method of making a project look professional every time it is compiled. Initially, time is required to learn your way around it, but it is worth it. Proofreading a document is no longer a ten-minute job, but rather a process involving a few days of bringing about a number of varieties to your document until there are only hours left to the submission deadline. There was no choice, but to stop.

Appendix B

Pictures of Factory

Figure B.1: Image of a full bin of bobbins.

Figure B.2: Image of the winders with the bins catching the bobbins.

Figure B.3: Image of 1 of 8 sections of a creel on top of a loom.

Figure B.4: Image of 1 of the 5 looms at MONN. The looms are under large frames which hold the creels above.