Bachelor Degree Project

Input Data Analysis for Detailed Flow Simulation of Manual Assembly Lines

Table of content

List of figures

List of tables

1. Introduction

Introduction describes the background of the company where the thesis was made. The goal is to understand what problem description of the thesis work is about and its aim and objectives. Furthermore, sustainability and thesis disposition will be presented at the end of this chapter.

1.1 Background

In the project the company is interested in the study of the losses of the production line and in the search of a model that simulates the real behaviour of the line, to be able to predict future reactions to the different changes that needs to be applied. It also should be generated a standardized model that they can use and imitate in their future projects. But there is an added problem, which is the continuous change of the assembly lines to adapt to the needs of customers, the human factor so different between each worker and the problems external to the line, such as the lack of material.

1.2 Problem description

The main goal of the project is to study the behaviour of an assembly line in order to build a simulation model with several causes of disturbances. The next step is to build a simulation model using more detailed data and compare the outcome with the results from the actual production line, and a model with a lower detail level supplied by the company The focus area is from the first station in a final assembly module to the last station in that module (after work station). The incoming materials that impacts on the performance of the assembly line should be analyzed and added to the model.

1.3 Aim and objectives

1. Define in the different ways the data that the company provides and analyze what data should be used as input data.
2. Deliver different models that are valid against the real assembly line and compare them with each other.
3. Describe which losses that could be important to add in order to build other final assembly flow simulation models that mimic the behaviour of a real final assembly line. Some examples of losses are material handling, quality, operators, and equipment problems. All losses that affect the assembly line.
4. Find the bottleneck of one of the final assembly flows.

1.4 Sustainability

Sustainable development can be identified as a struggle and need to take into consideration the welfare of our future generations. For this, it should be taken into account three basic pillars that will be developed in the course of this section. These bases are Social, economic and environmental.

As a keyword in this context, it is found, "Balance". For being able to have a sustainability development it is necessary to find the balance between the three categories mentioned above. In the point that concerns society, the ethics of each company and the people that are affected by the decisions of the companies should be taken into account. In addition to an international policy that preserves the ideals of the world protection, by creating laws that defend the environment and punish the abusive use of raw materials without respecting its replacement and the necessary times for it. The economic section may be the most difficult to describe because the market is completely bent on price competitiveness and where the customers exclusively seek the greatest monetary benefit, for these reasons the people should take refuge in a policy that not only defends the current welfare but that of the future people who will populate the earth. The environment may be the concept that is most related to sustainability. The reader has to understand each point described as a symbiotic system where each point contributes to the others and no one can prosper without the help of the other two.

The company defines itself as a component company focused on the human being, being a brand for people who care about other people and the world. Its philosophy is based on a Swedish word that is "omtanke" that has several meanings like "caring" and "consideration", as well as the importance of "to think again". As they study and analyze each time they make a decision like this, it will affect the world and the life of people.

The company are pioneers in responsible business, proudly forming part of the UN Global Compact.

In this project, it will be simulated the production line, with the objective of fighting against the different losses and obtaining the highest possible efficiency. The project will allow the company to detect the different losses, and the author will propose at the end some ways to fight this losses. The **economic** aim, it is very clear that if a higher availability and good throughput is achieved, the company will get a higher profit; and if the losses are decreased, they will not harm the economy of the company.

On the other hand, in the **social** section, if a reduction in the production costs is managed, it will be possible to lower the price of the product by being accessible to more people, in addition to increasing efficiency the factory will meet the demands of their clients. In the **ecological** target, it can be seen that if the production efficiency is increased, eliminating at the same time the losses. The company will achieve a reduction in the energy consumption of the line for the same volumes of stock, reducing at the same time the waste of natural resources that the loss of material implies.

Therefore, this project means a lot of sustainable growth for the company, allowing them to prosper and take a step forward with their philosophy.

1.5 Thesis disposition

This chapter gives a general picture of what each chapter is about.

1. **Introduction**: Presentation of the context to the readers, explaining the background of the company and the thesis work, problem description, aim and objectives, and sustainability.
2. **Methodology:** Information about how the project was carried out and what different method means.
3. **Frame of reference:** A better understanding of the theories that are relevant for the thesis work.
4. **Literature review:** Case studies regarding the thesis work. In this case, it was about simulation, workers in the assembly line and bottleneck.
5. **Validation**: Allows to give veracity to the project comparing the simulated model with the real line.
6. **Simulation model:** Gives an understanding of how simulation models were made.
7. **Results and analysis:** This chapter shows in detail the results of each simulation model performed.
8. **Discussion:** Authors reflection of the project.
9. **Conclusion:** The key ideas and most important concepts about how it has been carried out, the failures obtained and the results extracted, and the benefits that the project brings in the future.

2. Methodology

To begin with the project it is necessary to have a plan on how to begin and end it. By inspiration of Banks (2009) and his 12 steps simulation study, a process plan was created that was adapted to the project as it can be seen in figure 1. The process plan in figure 1 shows all the steps to the end of the project. In the project, each of these steps will be explained.

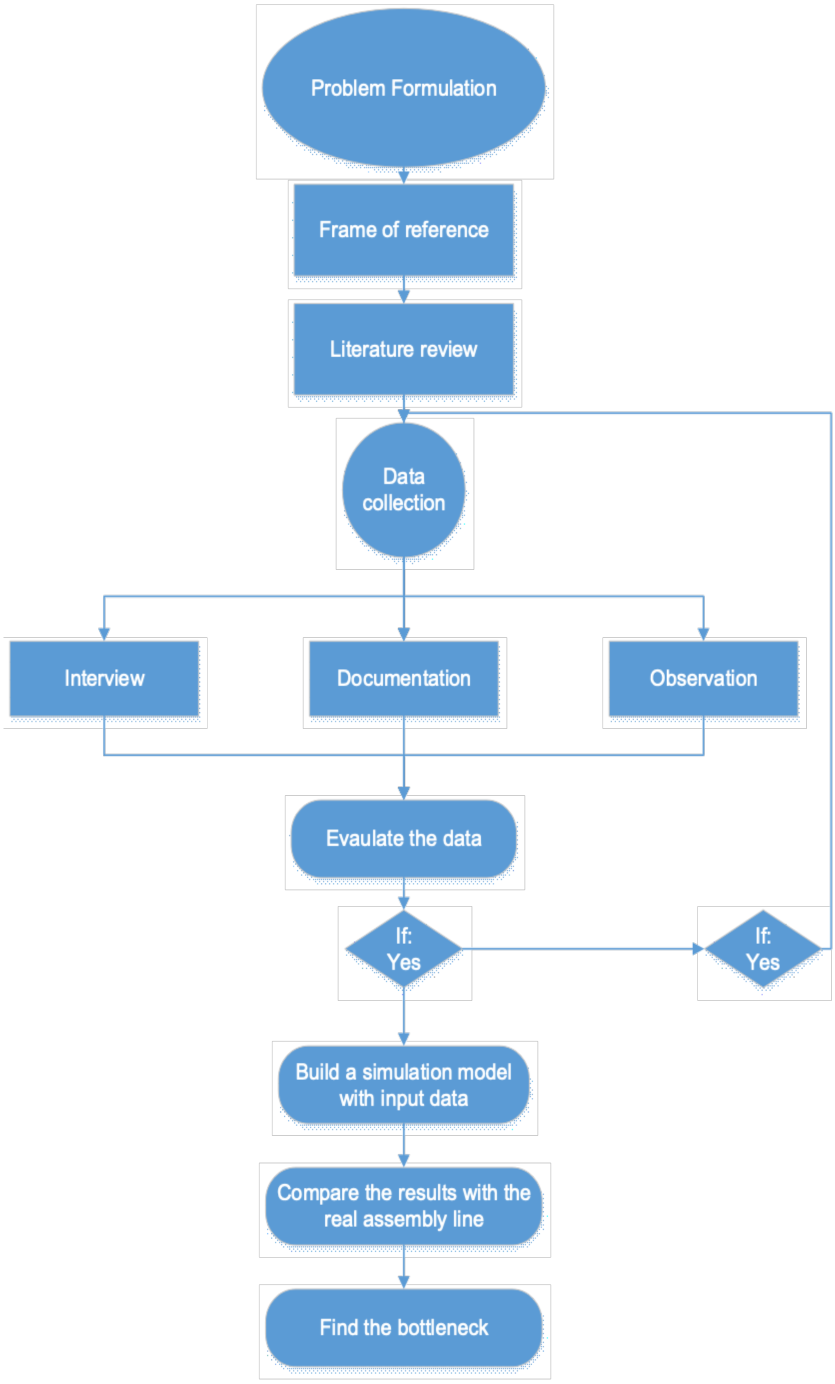


Figure 1. The process plan of the project.

Problem formulation: The beginning of the thesis work it is important to understand the objective/goal and delimitations of the project. This part is like a map that shows what to do in the project by following it step by step.
Frame of reference: Collection information of methods and tools in this project that were used.
Literature review: This section was used to understand about different areas that are related to this project by looking at old studies, for example simulation and bottleneck.
Data collection: In the project, this part is the most important task to do. It's because all the data that gets collected are going to be input data for the simulation model which will reveal if the model is valid with the real system or not.
Interview: Questions that could not be answered by looking at the data, were answered by production engineers or team leaders of the lines in the factory. It is important to have an open dialogue with people who know the system and have been working with it for several years.
Documentation: It does not matter if the project is small or big, documentation is always necessary to do. Document everything that could be important and it will be helpful later. It is an underestimated method that will bring success.
Observation: Observing a production system work makes everything easier to understand. Just looking at how operators work, how they rotate, how AGV's comes to the lines and etc. it will be simple to relate what data and what production engineers have to say.
Evaluate data: The data that was collected should be analyzed to separate the useful data from the rest. Besides, this will be analyzed furthermore to gain good input data for the simulation model.
Build a simulation model with input data: From the input data, the next step is to build a model that will hopefully have the same characteristics as the reality.
Compare the results with the real production line: When the model is built, it needs to be validated with the real production line. This will be done by showing the model to the people who are related to this project.
Find the bottleneck: The last step is to run the simulation model and analyze which station is the bottleneck of the system.

3. Frame of reference

This chapter will include simulation, collection of data, the bottleneck of the production line, integrated manufacturing systems and flexible workers. These topics are relevant for the project, therefore it is important to get an understanding of them.

3.1 Simulation

Simulation is designed to model a real system and analyzing the model to understand the behaviour of the system. The user could use simulation to construct the model and study the problem of the model by experimenting. (Pegden, Shannon & Sadowski 1995)
The simulation model can be tested to experiment with the system and observe how it would react.
Simulation could be used to planning, implementing and operating complicated systems/ processes. With simulation, the user could identify bottlenecks, find potentials of the system, see the performance of the system by doing tests and look into the minimum and maximum of utilization. (Steffen 2010).
Simulation has some advantages and disadvantages. Some examples of advantages are
It is possible to adjust the simulation time under investigation.

- Simulation study shows how the system operates
- There will be questions like" what if" which could be answered by simulation.

And for some disadvantages are:

- To build a model the user needs to have some training/experience from before.
- Modelling and analyses could take a lot of time and money. (Law 2007)

Systems could be two different types, one is discrete and the other one is continuous. Discrete system means that state variables could change instantaneously at some separated points in time. State variables of continuous system could change continuously over time. There are not many systems out there that could either be totally discrete or continuous, but it is possible to categorize the system to be either discrete or continuous because it is always one type of change that predominates. (Law 2007).

Law (2007) shows a way to study a system which is illustrated in figure 2.

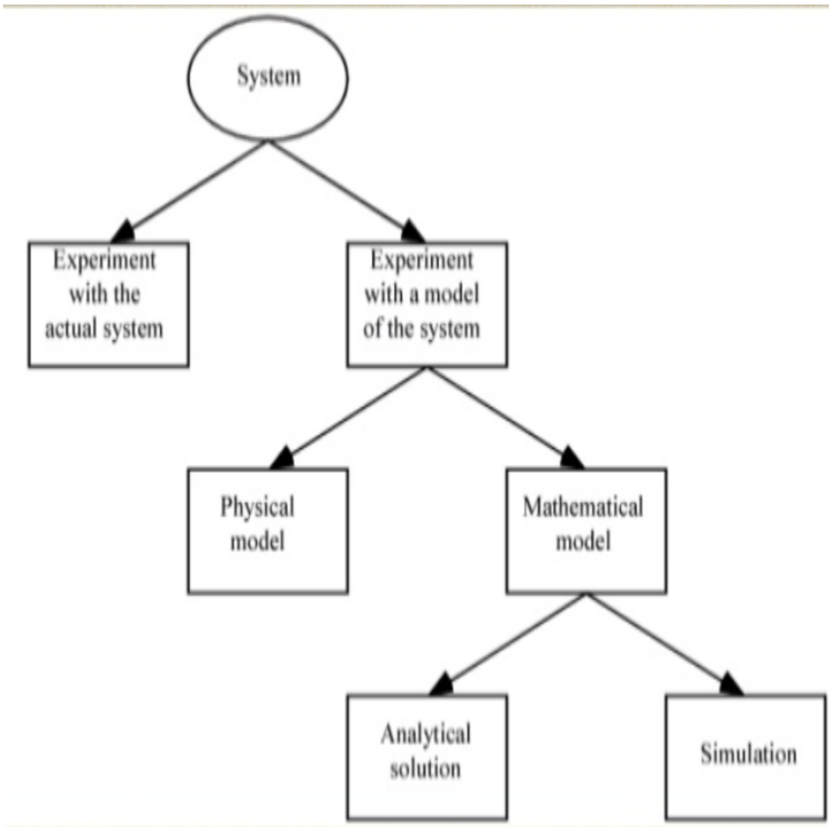


Figure 2. Laws model (2007)

One of the ways to build a system is coming from Banks (2009). The guide is illustrated in figure 3:

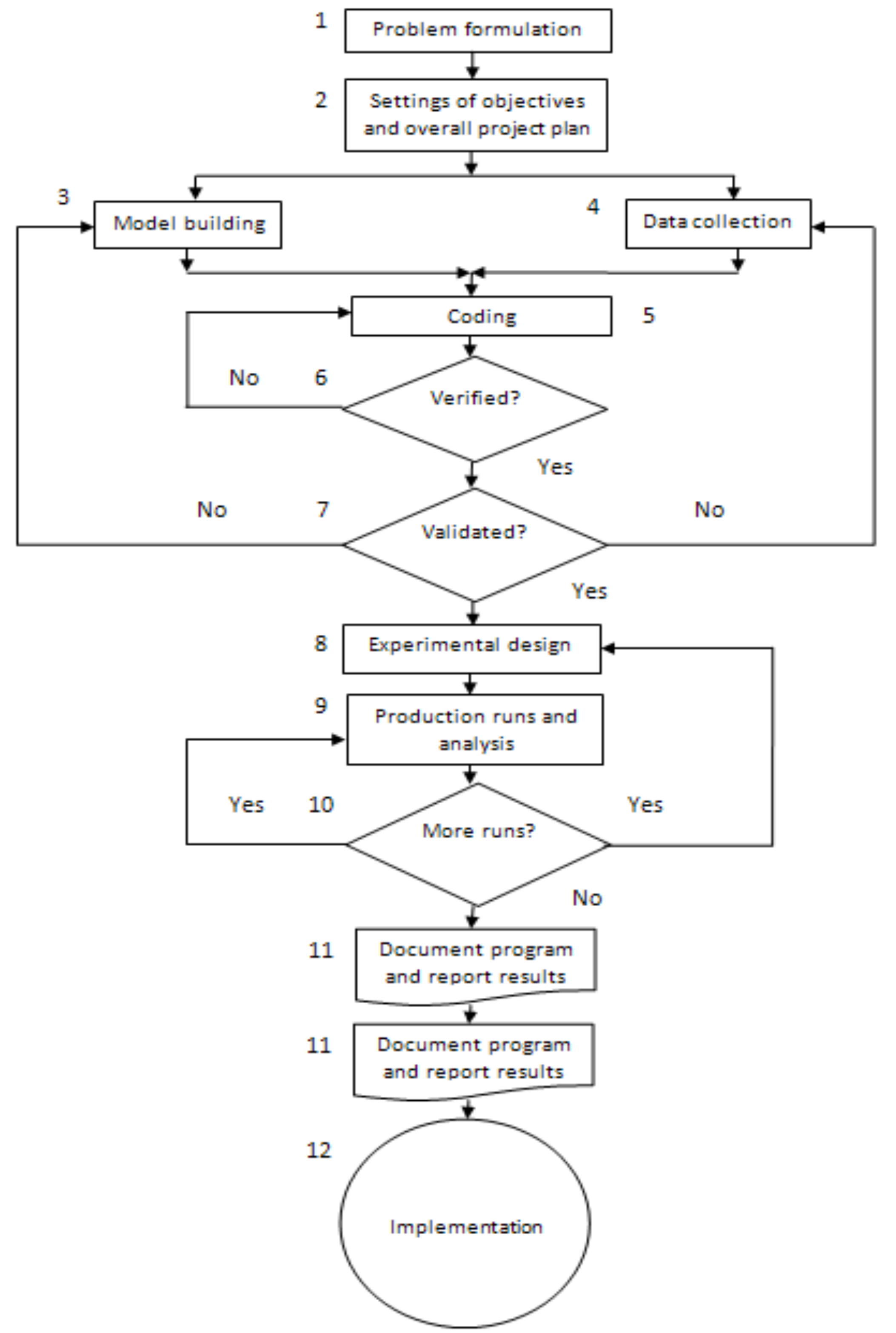


Figure 3. Banks model (2009)

Banks (2009) also explained what each step includes:

1. Problem formulation: Problem formulation is a place to identify a problem. If the policymakers have a problem or someone else then the problem must be described and explained very well by the analyst. If the problem comes from an analyst then it's important that the policymakers understand and agree with the formulation.

2. The setting of objectives and overall project plan: There are some questions that need to be answered by simulation. It should also be decided if the simulation is the right tool to use for the problem as formulated. Speculate the simulation is the best option, the overall project plan should include a number of people, the cost of study and the time to accomplish each phase of work.

3. Model conceptualization: It's very hard to build a successful and true model every time. The best way to start with modelling is to start with building an easy model and then continue to build a more complex model. After all the model doesn't need to be more complicated than that required to accomplish the goals for the model. It is recommended to be with the model user in model conceptualization because it will enhance the quality of the resulting model and more understanding of the model.

4. Data collection: Collection of data is very important when it comes to building a model, for example, the more complex the model gets the more data needs to be collected. The data collection should begin as early as possible because it takes a lot of time.

5. Model translation: Real-world systems result in a model which needs big information storage and computation, the model must be in a format where the computer can recognize. It is a requirement to decide where to program the model, in simulation language or simulation software.

6. Verified: Verification is something for the program which will be used in the simulation model to see if the program is acting as it should. If the program is good in the simulation model then the step is finished.

7. Validated: Validation shows if the model is good enough against the actual system and also it shows the difference between systems (real system and simulated system).

8. Experimental design: This is a step to determine the alternatives that should be simulated. Every time a system design is simulated there must be a decision about the length of the initialization period, the length of simulation runs and the number of replications to be made every time it runs.

9. Production runs and analysis: This step is used to see the performance of the system designs.

10. More runs: When the runs are done, the analyst will decide if there is a need to add more runs.

11. Documentation and reporting: Documentation has two parts: program and progress. Program documentation is very important for people to understand about how the program works, the program is going to be used again and it's good to have everything saved and also policymakers and users could make decisions by looking at the analysis from before. The reports should be updated frequently every month so that even those who are not involved in the operation often have the chance to follow it up by understanding the report. Every result in the report should be written well explained. This

is important for the decision maker because he/she will analyze the final formulation, the different systems that were named, experiment results and solutions to the problem.

12. Implementation: These previous 11 steps are the key to have a successful implementation phase, it's all dependent how good these 11 steps were. It is also important to know how much a model user knows about the model and its outputs and how often the model user was involved in the simulation process.

3.2 Collection of data

To collect data could be one of the most important and toughest things to do in order to solve a problem in the simulation. Before collecting data, it must be relevant and correct for the simulation model. (Banks 2009**).** If the input data is not good enough then the simulation results will not be as expected. So building a simulation model is dependent on the input data. Data collection requires a lot of time, it is recommended to begin to collect data as early as possible.

There is a difference between input data and output data. Input data is something that cannot change even if the system is improved but output data is the performance of the system which could change if there are any improvements to the system. (Banks 2009)

3.2.1 Input data

Input data is the key to a valid and real-life simulation model.

That is why it is extremely important to be critical and picky on the input data. Banks (2009) mentions four steps to create a simulation model with good input data.

1. Only collect data from the chosen system: This takes time and energy for data collection. If it is impossible to collect data because of the time, the law or there are no input data to collect, the expertise needs to give their estimations of the data.

2. Identify a probability distribution that shows the input data: As soon as the data collection is done it is time to make a histogram of the data. With the histogram, it will be easier to remove data which is not suitable.

3. Decide what parameters define the distribution family.

4. Evaluate the picked distribution and the associated parameters by analysis: If the data is not good enough, then the analyst could go back to the second step and choose a different probability distribution and repeats the steps.

Mchaney (1991) talks about how the input data is related to simulation, garbage in - garbage out. Doesn't matter how much data the analyst has, the simulation will not be improved if the data is not reliable. Mchaney (1991) speaks also about five ways to gather reliable input data:

1. Observation: Collecting data by observing the system/process.

2. Estimation: If there is no system to get data from, it's necessary to estimate input data.

5. Interpolation: If there is a model that already exists, by observing the model the data could be collected and interpolated into a model to get the same characteristics as the real one.

6. Expert opinions: Input data that are available will often be based on the opinion of experts.

7. Projections: The collected input data is derived from future projections.

3.3 Bottleneck of the production line

There are some definitions of a bottleneck in production. The bottleneck is the most active or working machine with the highest utilization in the line and/or is the machine with the longest average active period. (Bernedixen 2018).
This effect could cause the overall performance of the system which means that because of the bottleneck the system doesn't have the ability to work in the normal condition. According to Johannesson (2013) bottleneck restrict the capacity of the production system, even if the bottleneck is in somewhere else it would affect the production system if it's, example materials. Not only would it affect the capacity, but it would also be a big problem for the throughput (Hopp 2008).
Because of the restricted production flow, it's important that bottleneck stays alive all the time and have all the material it needs. Before products go into the bottleneck machine they should be tested to see if there are any defects on them, otherwise, it would be a waste of money and time to produce defected products on the bottleneck station. The bottleneck will be the one that's in charge of the whole system which means that other stations will be waiting with high capacity. (Johannesson 2013)
These bottlenecks could be categorized into three different types: Simple bottleneck, multiple bottlenecks, shifting bottleneck. A simple bottleneck is often only one bottleneck, one machine that affects the system during a period. The second one is multiple bottlenecks which means there are more than one machines that affect the system but this time, they are fixed for the entire period. The last one is called shifting bottleneck because the bottleneck is shifting between one station to another all the time and there is no bottleneck for the entire period. (Lima, Chwif & Barreto 2008)
There are many ways to identify/find bottlenecks with different kind of methods and tools for example simulation and value stream mapping but not only it's important to find it, but it should also be eliminated for life. Bicheno (2009) talks about a method called the theory of constraints (TOC) where he says, trying to balancing the capacity by sorting products that come into the flow is not a correct approach because the production flow could be faster and slower. Instead, it's better to look into a continuous flow. It means that if no product needs to wait or the process needs to stop or be blocked. Everything is going smoothly.

3.4 Integrated Manufacturing Systems:

In this point, it is going to be discussed the different I*ntegrated Manufacturing Systems,* which consist of several Workstation/or machines that are connected to a handling system that transports the products between these machines in a certain order of processing.
Most of these systems are managed by computer control to measure the times taken by each machine or worker to process the product and to coordinate the equipment and the stations between them. Hence, the main components of a system are the workstations/or machines, material handling equipment, and computer control. it should not be forgotten to mention the workers responsible to operate each individual station or the ones who are in charge of supervising on overall the system. (Groover, 2010).
It could be found different integrated systems like manual and automated production lines, manufacturing cells or flexible manufacturing systems (currently, the last one mentioned is

the most popular strategy and a large part of the companies are trying to reconvert their industries into this type).
Material handling means the movement, storage, protection and control of material throughout the distribution process. This term is intimately related to the activities that take place inside a facility. Since the products should be moved within the industry to be converted into his final state. Material handling includes three kinds of activities (1) loading and positioning products at stations, (2) unloading work units from stations, and (3) transporting products between stations. Eventually, material handling is used as a temporary storage function, creating a queue of products, so they are ready for fast processing without idle times. (Groover, 2010).
Two main categories of material transport equipment can be differentiated according to the kind of industry and its plan: fixed and variable. The difference between them is easy, the first one is orientated to a single-product industry without no customization, instead of variable routing where the works units can move through the process taking in consideration the number of variants, or the features that the customer has chosen.
As can be appreciated, the type of industry defines the type of handling system that is used. In this case, since the industry studied is based on the product, within an advanced sector such as automotive, they have AGVs. These AGVs are programmed to follow a single path, so the classification for this case of the material transport equipment is fixed routing. (Groover, 2010).
With all the information that has been explained at this point, it is easy to check how production lines are really an important class of manufacturing system because they allow creating huge volumes of similar products. The lines should divide the total work of the product into different steps that must be carried out in the simplest and fastest way.
The reader should remember that the rate of the line is limited by its slowest station. Workstations whose rate of working is faster than the slowest one will be limited by that bottleneck station. (Groover, 2010).
This way of acting can be defined as the theory of constraints (TOC), which says that "a chain is no stronger than its weakest link"

3.5 Flexible workers

The production systems are becoming more flexible and it’s important to manufacturing companies to handle the demand with a high variation of products with fluctuant volumes (Małachowski & Korytkowski 2016).
Each worker has an availability time and standard processing time for all tasks in each station are established and deterministic. It would be trouble for the production line to reach the cycle time and takt time where the operators are not flexible enough and inexperienced. Companies need to provide cross-training to help workers to develop multi-tasking skills. (Lian, Liu, Li & Yin 2018).
Companies are benefiting from multi-skilled workers because of the lower number of staff. Multi-skilled workers could be hired easier for jobs. These multi-tasking workers will increase the flexibility in the production when variation in demands. (Małachowski & Korytkowski 2016)

3.6 Takt time and continues flow

In lean production, the takt time is a fundamental concept to get a simultaneous production flow from raw material to a finished product.
The takt time is the speed of the production system and that is why it is necessary to understand it, otherwise, it would be hard to analyze any operations or create a stable one. (Bicheno 2009).
The way to calculate the takt time is to divide the production time in one day with the takt. Takt is the average demand. And often people mix this term with cycle time.
The big difference is that the takt time is based on the demand while cycle time is about how fast could a machine/operator produce or finish their task. Besides, the cycle time should not exceed the takt time or else a bottleneck will appear. (Peterson 2009).
So, the demand is the one that determines the takt time which means that if the demand varies often then the production line needs to be balanced and the operators need to be flexible. But the ideal idea is to have the flow as smooth as possible. (Bicheno 2009).
If the production flow/line is moving by the takt time, it will be easy to find out where the problem is if the flow stops suddenly, furthermore the deviation will come up to the "surface" and it can be eliminated.
Another way to see if there is a waste in the production flow is, if the product did not reach the customer within the takt time then there is a waste somewhere. (Peterson 2009).
The takt time affects also the rest of the company, for example, leaders and bosses in the organization. They want to know the results so they can have future plans for growing as well as making the right decision at the right time. (Peterson 2009).

The word continues flow appears quite often, so what does it mean? The right answer is the material and product are always in motion without any stops.
Every stop could be waste lite waiting time, longer lead time and it leads to less flexibility and waste of money. As everyone knows it is very hard to achieve a continuous flow without any stops, but trying to have as fewer stops as possible is a good start to get a continuous flow. That is a goal that many companies trying to accomplish.
Another way to get close to the goal is to shorten the distance between operations, have small buffers and transports that come often. (Peterson 2009).

4. Literature review

This chapter presents researches on similar studies regarding this study. At the end of this chapter, there will be a part named analysis of literature review. Two big areas in this project are about simulation, bottleneck, and workers in the assembly line, for that reason the focus is on them. The literature review summarized each study and shared the most important of input data on literature review.

4.1 Simulation

Nagi, Chen, Wan (2017) performed a study where they used simulation modelling and analysis to improve line performance. On the beginning, the tool value stream mapping was used to get an understandable view of the line. Afterward a Yamazumi chart was created to see the capacity of the line with all the stations. The simulation was used to create a module that has the same characters as the real assembly line. The analysis showed that the bottleneck station 17 was the problem for the whole line. Via simulation, bottleneck analysis and cycle time study began on station 17 to reduce the cycle time. The results came back and the new cycle time was added in the simulation module which could tell how much the performance was improved and how good WIP limits values were with the design of experiment (DOE).

Danielsson (2016) did a study where they analyzed how a flexible assembly line could manage two different products in the same line. This flexible assembly line will be used to manufacture different products in the future. Information about the system was collected before creating simulation models in the software plant simulation. The models were randomly simulated to investigate if models could handle varies in demand. Afterward, the bottleneck was identified and an improvement was proposed. In total it was six studies that were experimented in the simulation to identify how many operators needed to be in the assembly line to handle the variations of the demand.

Malaki (2012) did research about an assembly line that will be developed into a better line by conceptual design. The main focus of the research is about finding possibilities to improve throughput without any investments. The assembly line is comprised of 40 assembly stations where products are fixed on the customized pallets which are moved between stations. In this assembly line workers are moving instead of products, this means that workers are getting one product to assemble from station one to the last station and then moves back again to station one or the beginning of the assembly line. So Malaki (2012) build a simulation model of the assembly line by using Plant Simulation. All the data and other parameters were included in the model. By analyzing the results from the simulation, the improvement would be possible by arranging workers with different variability in the determined patterns.

Yesigul & Nasser (2013) did a thesis work about creating a simulation model for assembly lines, find the best-operating conditions and improving the productivity of the line by creating solutions for problems. It all started with examining the whole assembly line to get an understanding view, for example how many workers, what kind of problems just to appear, what the throughput is and how many stations. Dividing the production line into eight stations made it easier to calculate the cycle time, waiting time and other parameters that would include in the simulation model. When the model was built, it ran with different kind of scenarios where the number of workers were changed, the number of pallets and distribution of workers. The conclusion of the project was to start with continuous improvements in the factory to get a continuous flow but the production line had two big problems, lack of material stops the whole line and the second is workers needs to have more training about production line to reduce the time of repairing and a number of shutdowns.

4.2 Bottleneck

Kamsemset and Kachitvichyanukul (2007) did a study that shows how to identify bottleneck stations by simulation. For this study the concept theory of constraints (TOC) which has five steps:

1. Identify the system constraints
2. Benefit from the system constraints
3. Subordinate the system constraints
4. Elevate the system constraints
5. Repeat the process from the first step if it was not good enough

TOC shows the output rate of a system that is limited by the slowest machine. For the bottleneck identification, there were three things that were taken into account: Data collection of a process and machines, utilization factor and bottleneck rate. (Kamsemset and Kachitvichyanukul 2007).
Identifying bottlenecks there are some criteria that need to meet:

1. High value of the machine and process utilization
2. High value of the processor utilization factor
3. Low value of the product bottleneck rate.

The study showed two different example cases about a bottleneck.
The first case was about how to identify a bottleneck by using simulation. Collected data were analyzed where G identified as a bottleneck because it meets all of the criteria for a bottleneck. Also, the G machine would be improved then the system performance would also be better. (Kamsemset and Kachitvichyanukul 2007).
The second case was about multiple bottlenecks. This time the machine E was identified as bottleneck but improving the only E in the system wouldn't improve the throughput (TH) of the system which means that there are more machines that could be bottlenecks. After rerunning of the simulation, machine A identified as a bottleneck and now as result E and A could improve the system. (Kamsemset and Kachitvichyanukul 2007).
Ali, Seifoddini & Sun (2005) did a study in a assembly line where the stations are unblanaced. The output got improved when the bottleneck got identified and eliminated. This work was done by building a simulation model with all the necessary data that was recuired. Not only was the output better, also the takt time which could meet the future production, now it is possible increase the production, effieciency got improved and flexibility for changes in demand.

4.3 Workers in the assembly line

As Alzuheri, Lounge and Xing (2010) stated the traditional manual assembly line was modelled where every worker have one station to work on and the assemble parts are transported to workstations so workers do not move at all, this is fixed workers (FW). Manual workstations are used in the assembly line when the work is too complex or when the demand is unstable and it's expensive with flexible machines. In the assembly line with FW, the production rate is affected by slowest worker and workstations that have long process time and balancing the assembly line could also affect workers speed and skill.
Alzuheri (2010) mentions that walking workers (WW) modelled the way that workers assemble a product from the beginning to the end by walking in the line. When WW is applied in the production line there will some benefits, for example, it will be easier to balance the line, better variations in work time, decide how many workers are needed in the line depending on the demand and the cost will be lower for company. The negatives with WW is workers get too much pressure because of the highly demanding time where they working with multitasking which includes repetitive motion movements. The result from this could harm operators for example injuries, accidents, sickness, and bad output.
Wang, Owen, and Mileham (2005) made a study about FW and WW and they affect the efficiency of the line comparing to each other. The study showed that the FW line continuously loses efficiency when the line grows bigger. With WW is the other way around which means that the efficiency is increased when the line grows. As a result shows, it would be beneficial to use WW instead of FW because WW line would make the output and efficiency higher. But in the end, the slower worker in a working line could affect the output of the production line.

4.4 Analysis of literature review

The literature review chapter presented topics regarding simulation and bottlenecks. This chapter gives an understanding and knowledge about methods that will be used in order to complete the project. Most of the studies in this chapter are about simulation, bottleneck and walking workers. Each study has a lot of focus on the beginning of the project, example understanding the project, planning part, information and data collection and what kind of methods that were used. The author, name of the study, the date and the link of the study is shown in the chapter references. Summarize of the chapter:

- Before building a simulation model, it's important to analyze the production system and collect reliable data.
- The simulation model is used to find improvements and bottlenecks.
- Statistics show that walking workers are much better than fixed workers overall.

5. Case Study

In this chapter, the authors will explain in detail the software used, how the data has been collected, the numerous failed attempts, and the other successful ones; the implementation of the correct and the learned of the wrong; how transforming raw data into accessible and useful data , each line that has been programmed and in general the reason for each small detail of this project.

5.1 Plant Simulation

This first subchapter will speak about why this simulation software, awarded by Siemens, has been chosen. The simulation software of Tecnomatix Plant Simulation plants allows the simulation and optimization of production processes and systems. Through Plant Simulation you can optimize the flow of materials, the use of resources and logistics at all levels of plant planning, from global production facilities to local plants or specific lines.
Plant Simulation can build digital environments, where can be created models that fully replicate the behaviour of the real line, understanding and being able to change all its characteristics to obtain the best performance. Thanks to this software, it is allowed to carry out all kinds of experiments or modifications in the lines without the risk of suffering any loss or any cost, even before building the lines it would be possible to plan them.
These characteristics are generic and common of all simulation software such as Arena or Flexsim. The reason why this particular software has been chosen has been due to several characteristics:

- Its broad programming language Simtalk, which allows to edit and customize the behaviour of the line according to our parameters, this programming language is object-oriented, in addition to fulfilling an inheritance and a hierarchy that allows breaking down the most difficult tasks into diverse simpler.
- The opening to import data from other systems, such as databases (Access or Oracle) or spreadsheets of programs such as Excel.
- Integration within the industry, allowing the reading of data directly from the PLM, or the use of data directly from the layout as Autocad.
- Finally, it should not be forgotten to mention the tools of detection of problems (Bottleneck Analyzer or the study of Diagrams of Sankey), besides having many others to fix these problems (Genetic Algorithms or Neural Networks).

As mentioned in the characteristics, the use of this program was essential to develop a model that acted faithfully as the real production line, since the company already had a too simplified model of little utility developed in FACTS.

5.2 Assembly Line Description and Production Flow

The model is focused on the simulation of an assembly line, which is dedicated to the production of three different types of components. The module consists of 18 assembly stations, divided into three lines of 6 stations each. As can be seen in the map, the layout of the work stations in each line is 3x3 forming an U. The assembly line is shown in figure 4.

The complete module has a total of 33 AGVs distributed in buffers of:

- Five at the beginning of the module
- Six in each line
- Three as a buffer between lines
- Four at the end of the module

The line has one worker per station as usual. Although the demands can fluctuate and sometimes reduce one worker per line, leaving a total 50% workers in total. Although the module is designed to reduce even some more workers per line but has never had to do so due to high demand. In the following sub-chapters, a comparison of the final throughput will be developed according to the number of workers.
The module works synchronized with a "Takt Time". This means that no station can start a new process without waiting for the rest of the stations to finish their corresponding work. The "Takt Time" can be considered as the time it takes to perform the work plus the time of transporting the product. Whenever the product is moved the worker must accompany him to continue the work at the next station.

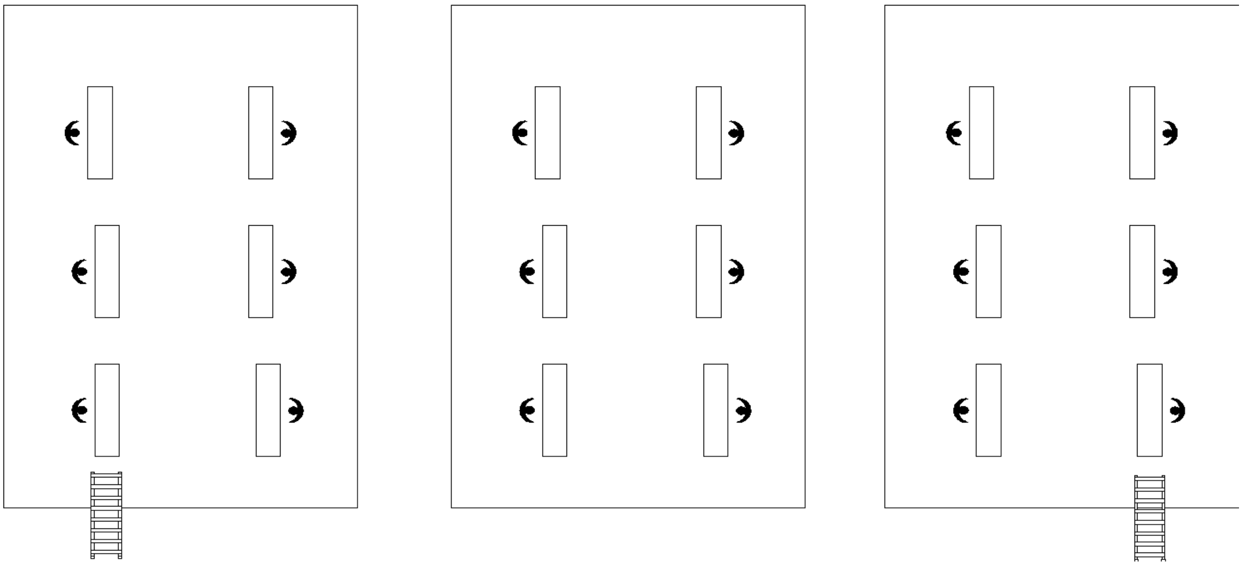


Figure 4. Assembly line

5.3. Data Collection and processing

After having selected the software that will be used and having set the limits and objectives of the project, it will be necessary to move to the factory for the data collection of the hand of the Production Engineer responsible for the module 4 that is going to be simulated.

5.3.1. Stations Time

The production engineer and leader of the module provided the data in this type of tables.

Log Type	Cycle Start	Cycle End	Difference	Area	Station	Operator	Pace
STATION	2019-01-09 15:18:49:00(	2019-01-09 15:25:48:000	-6148	MOD04_2	04S250	5	111
STATION	2019-01-10 15:10:18:00(	2019-01-10 15:15:14:000	-9529	MOD04_2	04S250	9	111
STATION	2019-01-10 18:24:46:00(	2019-01-10 18:28:20:000	-8642	MOD04_2	04S250	6	111
STATION	2019-01-10 07:09:48:00(	2019-01-10 07:13:21:000	-16197	MOD04_2	04S250	13	111
STATION	2019-01-10 16:26:14:00(	2019-01-10 16:29:31:000	-10364	MOD04_2	04S250	4	111
STATION	2019-01-08 18:39:44:00(	2019-01-08 18:42:42:000	-15135	MOD04_2	04S250	26	111

Table 1. Times of the production line.

These data in table 1 can not be entered in the program, and it is even difficult to obtain some type of information at a glance from them without prior processing. Therefore, new data will be calculated, taking into account the future values to be introduced in Plant Simulation, to be used as input data has been helpful in Excel software and the tools it provides. As can be observed, there is a column that identifies the cycle start time and the end time of the cycle. From these two columns, a macro has been programmed that will be left for future use, which extracts the time of each cycle in two other columns with a sexagesimal numerical format that may be useful. The "processing" column is the subtraction of "CycleEnd H" and "CycleStart H", in this way all the times used to perform a job in a station are obtained; A new format has been applied that shows all the data in seconds. In the following table 2, its possible to see the three new columns highlighted:

Log Type	Cycle Start	Cycle End	Cycle Start HOURS	Cycle Start HOUR	ProcessingTime	Difference	Area	Station	Operator	Pace
STATION	2019-01-09 15:18:49:00(	2019-01-09 15:25:48:000	15:18:49	15:25:48	419	-6148	MOD04_2	04S250	4	111
STATION	2019-01-10 15:10:18:00(	2019-01-10 15:15:14:000	15:10:18	15:15:14	296	-9529	MOD04_2	04S250	11	111
STATION	2019-01-10 18:24:46:00(	2019-01-10 18:28:20:000	18:24:46	18:28:20	214	-8642	MOD04_2	04S250	4	111
STATION	2019-01-10 07:09:48:00(	2019-01-10 07:13:21:000	07:09:48	07:13:21	213	-16197	MOD04_2	04S250	27	111
STATION	2019-01-10 16:26:14:00(	2019-01-10 16:29:31:000	16:26:14	16:29:31	197	-10364	MOD04_2	04S250	10	111
STATION	2019-01-08 18:39:44:00(	2019-01-08 18:42:42:000	18:39:44	18:42:42	178	-15135	MOD04_2	04S250	27	111
STATION	2019-01-08 21:36:26:00(	2019-01-08 21:39:23:000	21:36:26	21:39:23	177	-18545	MOD04_2	04S250	1	111
STATION	2019-01-10 02:05:38:00(	2019-01-10 02:08:15:000	02:05:38	02:08:15	157	-19845	MOD04_2	04S250	10	111
STATION	2019-01-10 18:50:47:00(	2019-01-10 18:53:12:000	18:50:47	18:53:12	145	-3426	MOD04_2	04S250	22	111
STATION	2019-01-08 09:47:35:00(	2019-01-08 09:49:55:000	09:47:35	09:49:55	140	-10356	MOD04_2	04S250	18	111
STATION	2019-01-10 12:35:13:00(	2019-01-10 12:37:32:000	12:35:13	12:37:32	139	-12212	MOD04_2	04S250	4	111
STATION	2019-01-07 16:41:24:00(	2019-01-07 16:43:31:000	16:41:24	16:43:31	127	-6252	MOD04_2	04S250	24	111
STATION	2019-01-07 14:31:48:00(	2019-01-07 14:33:42:000	14:31:48	14:33:42	114	-7405	MOD04_2	04S250	23	111
STATION	2019-01-08 23:27:03:00(	2019-01-08 23:28:56:000	23:27:03	23:28:56	113	-8268	MOD04_2	04S250	11	111

Table 2. Cycle times of the production line.

The data shown has been manipulated and modified in order to keep the company's confidentiality.

```
Sub ExtractTimes()
'
' TIMES Macro
'
```

```
' Keyboard Shortcut: Ctrl+u
'
    Application.ScreenUpdating = False
    Application.Calculation = xlManual
    Columns("D:D").Select
    Selection.Insert Shift:=xlToRight,
CopyOrigin:=xlFormatFromLeftOrAbove
    Columns("D:D").Select
    Selection.Insert Shift:=xlToRight,
CopyOrigin:=xlFormatFromLeftOrAbove
    LR = Cells(Rows.Count, 1).End(xlUp).Row
    For i = 2 To LR
      Cells(i, 4) = Mid(Cells(i, 2), 12, 8)
    Next
    For i = 2 To LR
    Cells(i, 5) = Mid(Cells(i, 3), 12, 8)
    Next
    Range("D1").Select
    ActiveCell.FormulaR1C1 = "Cycle Start HOURS"
    Range("E1").Select
    ActiveCell.FormulaR1C1 = "Cycle End HOURS"
    Columns("D:E").Select
    Selection.NumberFormat = "[$-F400]h:mm:ss AM/PM"
    Columns("F:F").Select
    Selection.Insert Shift:=xlToRight,
CopyOrigin:=xlFormatFromLeftOrAbove
    For i = 2 To LR
    t1 = Format(Cells(i, 5), "hh:mm:ss")
    t0 = Format(Cells(i, 4), "hh:mm:ss")
    Cells(i, 6) = Format(TimeValue(t1) - TimeValue(t0), "hh:mm:ss")
    Cells(i, 6).NumberFormat = "[ss]"
    Next
    Range("F1").Select
    ActiveCell.FormulaR1C1 = "ProcessingTimes"
    Application.Calculation = xlAutomatic
    Application.ScreenUpdating = True
    End Sub
```

Figure 5. Excel macro, dividing data.

The macro shown in the figure 5 above allows the extraction of the second and third column of the data sheet provided by the company in a format of HH: MM: SS; this macros also

subtracts these two columns and saves that result in a new column that will be very useful since they are the processing times of each assembly.
A brief description of the code will be made; row 7 and 8 highlighted with a circle, will be written in each generated macro to optimize the speed of the program. At the same time, this configuration made to optimize the speed should be corrected to the predetermined values in the code lines 39 and 40.
The next paragraph wrapped by a rectangle generates 2 new columns where the values of column 2 and 3 will be extracted in a numerical format that allows the realization of calculations.
The next point, highlighted by a key, contains two loops in charge of going through each row of columns 2 and 3 and extract the useful values for the calculation of times in the new columns created.
The following paragraph marked by a cloud gives a name to classify the new two columns created.
The last key marks a loop where the character strings, of the data extracted in the two new columns, are formatted again and subtraction is performed with a sexagesimal numerical calculation and the result obtained is expressed in a new column called "Processing Times" in seconds format.
The data obtained after applying the macros are general of all the stations which is in figure 6, so each station has been divided into a worksheet in order to classify them and work them individually. Each station has its own processing time independent of the takt time that will be introduced in a basic model to corroborate the accuracy of the results. This operation has been carried out by means of other macros that taking only into consideration the normal times between 300 <T <541 seconds, executes a weighted average that can be considered as the processing time of the station. In this way, you get the results of a basic model that can be obtained using FACTS.

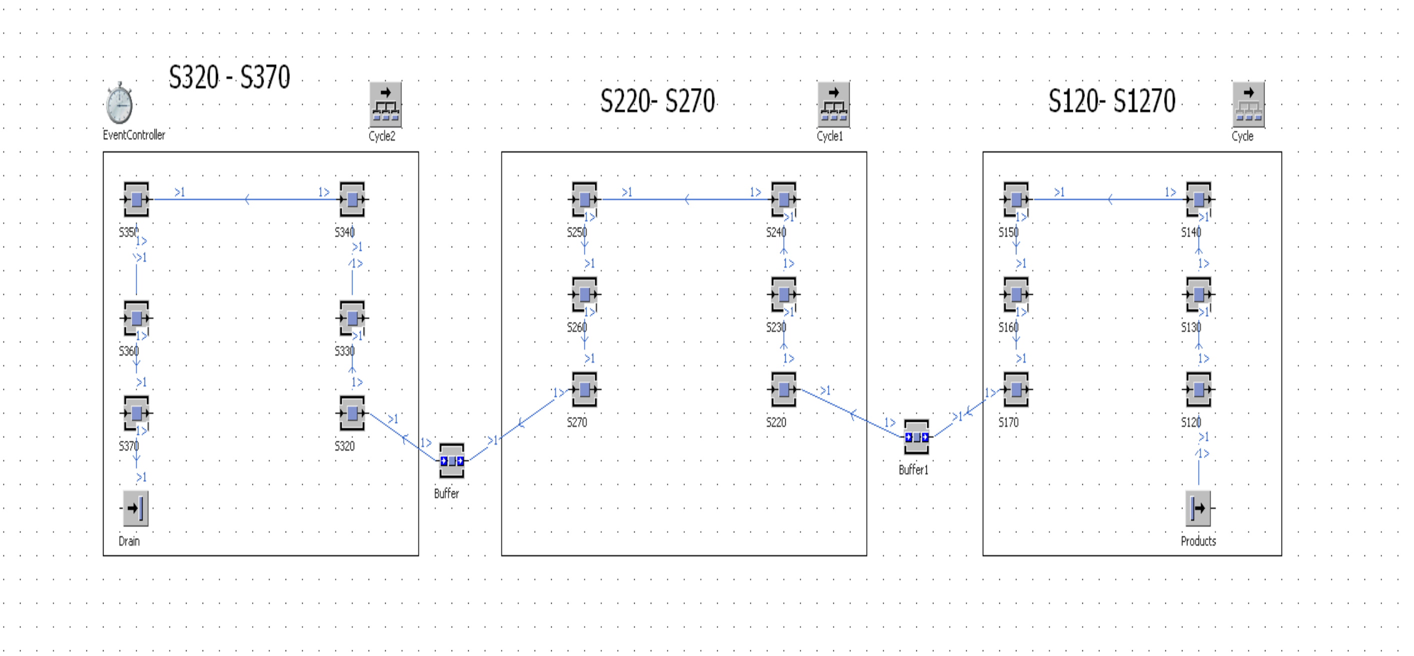


Figure 6. Simulation model

```
Sub ProcessingTimes()

    Application.ScreenUpdating = False
    Application.Calculation = xlManual
    Dim Row as long
    Dim x as long
    Row=2
    x=0
    Range("M:M").Delete
        Range("R4").Select
    ActiveCell.FormulaR1C1 = "AverageTimeStation"
    Do While Cells(Row, 6) <> ""
         x = x + Cells(Row, 6).Value
         Row = Row + 1
    Loop
    Range("S4").Value = x / (Row - 1)
    Range("S4").NumberFormat = "[ss]"

    Application.Calculation = xlAutomatic
    Application.ScreenUpdating = True

End Sub
```

Figure 7. Excel macro processing time.

As has been mentioned previously, the first and last lines of the code in figure 7 are dedicated to the optimization of the program. The first thing that is done is the creation of the variables that in this case will be of the "long" type, a type normally used as a counter because it allows storing numbers of up to 64 bits.

After the creation of the variables, the comments column is deleted because it will not have any use in the project. The name "Average Time Station" is added in cell "R4"; and a loop is executed to calculate the average of all processing times and the result is stored in the cell next to R4, S4. This last cell is modified with a new format in seconds.

After having calculated the basic values for the elaboration of a simple model in which an approach to the real results can be observed, a search will be necessary to obtain more data that allows more precision in the model.

To obtain the workers within the thousands of data that you have, a macro is a great help extracting the IDs of each operator:

```
Sub ExtractWorkers()
'
' Extract Macro
'

'
    Application.ScreenUpdating = False
```

```
    Application.Calculation = xlManual
    Columns("K:K").Select
    Selection.Copy
    Range("T1").Select
    ActiveSheet.Paste
    Application.CutCopyMode = False
    ActiveSheet.Range("$T$1:$T$7035").RemoveDuplicates Columns:=1, Header:= _
        xlYes
    Application.Calculation = xlAutomatic
    Application.ScreenUpdating = True
End Sub
```

Figure 8 Excel macro dividing workers time

As in the previous sub-programs that have been created, the first and last lines are aimed at optimizing the program.

This small subprogram in figure 8 allows the obtaining of all workers in a simple way without the need to count them one by one. Normally, among the more than the 7000 assembly times of each station, there is a total of about 54 workers that vary depending on the line they are working on. This will allow the classification of them in efficiency according to the average time of the accomplishment of their task in each station.

Now, at the moment that the IDs of all the workers have been obtained, the work will focus on the classification of the workers according to whether they are fast, normal or slow. To be able to classify them later according to their processing times in some parameterized distribution curves that will be introduced later in Plant Simulation to replicate their behaviour.

Operators' classification will consist in finding their average execution time in each station, and dividing them so that the 25% with the least processing time obtained will be fast, the next 50% will be considered normal, and 25% with more time will be slow. It has been done in this way, because the stations although they are designed to meet a takt time of 173,53 seconds perform different operations so the processing times will also be different, and you must have a sufficient number of operators in each group (fast, normal and slow), to have enough data to adjust the distribution curves.

```
Sub AvgProcTimeWorkers()
'
' AvgProcTimeWorkers Macro
'

'
    Application.ScreenUpdating = False
    Application.Calculation = xlManual

    Range("U2").Select
    ActiveCell.FormulaR1C1 = "=AVERAGEIFS(C6,C11,RC[-1])"
    Range("U2").Select
    Selection.AutoFill Destination:=Range("U2:U73"), Type:=xlFillDefault
```

```
    Range("U2:U73").Select
    ActiveWindow.SmallScroll Down:=-69
    Range("U1").Select
    ActiveCell.FormulaR1C1 = "ProcTimeWorkers"
    Range("U2").Select
    Range("U2:U73").NumberFormat = "[ss]"
    Application.Calculation = xlAutomatic
    Application.ScreenUpdating = True
End Sub
```

Figure 9. Excel macro Average Processing time.

This small program in figure 9 allows the extraction of the average times that each worker takes to execute a task in a specific station. For this purpose, the formula that incorporates the "Excel" program for the calculation of averages will be used and a condition will be applied to said formula so it only calculates the average of each worker in concrete, comparing it with the worker studied in the column of the left and do not run any calculation on the remaining workers.

As the number of workers in each station varies, this operation has been carried out up to a maximum of 73 workers since normally the number of workers does not exceed 60. As it could be observed in the other programs, a calculation is being made in seconds, a unit of time that works in the sexagesimal system, so again the corresponding format should be applied to the 73 cells.

Finally, the efficiency of each operator has been found following an inverse proportionality. This calculation, although it has been unnecessary later in the project, was thought with the objective of creating a simulation model where the workers were replicated by real people and not as objects with distribution curves according to their behaviour as can be seen below.

```
Sub Efficiency()
'
' Efficiency Macro
'

'
    Application.ScreenUpdating = False
    Application.Calculation = xlManual
    Range("V1").Select
    ActiveCell.FormulaR1C1 = "Efficiency"
    Range("V2").Select
    ActiveCell.FormulaR1C1 = "=(R4C18/RC[-1])*100"
    Range("V2").Select
    Selection.AutoFill
Destination:=Range("V2:V74"), Type:=xlFillDefault
```

```
    Range("V2:V74").Select
    ActiveWindow.SmallScroll Down:=-54
    Application.Calculation = xlAutomatic
    Application.ScreenUpdating = True
End Sub
```

Figure 10. Excel macro efficiency.

This subprogram shown in figure 10 was created with the intention of using the object "Worker" predefined by plant simulation, instead of replicating the behaviour of the workers by parameterized curves depending on the type of worker that represents the pallet in the station.

Finally, that simulation model was not created, but the macro was left in Excel because it allows seeing in a simple way the efficiency of each worker in easy percentages depending on the type of station.

The calculation of the efficiencies was obtained with a simple mathematical calculation of proportionalities. The time average of all the assemblies of each station is taken as a reference of time, this time will be 100%. If it is known that a worker with a time equal to the processing time of the station has a 100% efficiency by inverse proportionalities (inverse proportionality has been mentioned, because the shorter the time a worker executes a task, the more efficient it will be) the efficiency of the rest of the workers can be known. This calculation is shown in the following formula E.Q.1:

$$Efficiency = \frac{ProcessingTimeoftheStation}{X} * 100 \qquad \text{E.Q.1}$$

Where "x" is the average time that each worker uses to execute a task

The operation of the code follows the formula explained above but applied to the 73 workers taken as reference in each station, substituting the "x" for the average time that each worker uses to make an assembly.

To be certain that the corresponding operators are placed in each group, all operators with a number of jobs per station less than twenty processes have been eliminated, that is, all those workers with less than twenty processing times will be removed from the classification in Fast, Normal, Slow, because it would falsify the data. To do this, again, help has been taken from macro in figure 11.

```
Sub DeleteWorkers20PT()
'

'
    Application.ScreenUpdating = False
    Application.Calculation = xlManual
    Range("W1").Select
    ActiveCell.FormulaR1C1 = "AmountPT"
    Range("W2").Select
    ActiveCell.FormulaR1C1 = "=COUNTIFS(C11,RC[-3])"
```

```
    Range("W2").Select
    Selection.AutoFill Destination:=Range("W2:W74"), Type:=xlFillDefault

    Range("X1") = "WorkersGOOD"
    Range("X2").Select
    ActiveCell.FormulaR1C1 = "=IF(RC[-1]>20,RC[-4],"""")"
    Range("X2").Select
    Selection.AutoFill Destination:=Range("X2:X74"), Type:=xlFillDefault
    Range("X2:X74").Select
    ActiveWindow.SmallScroll Down:=-48
    Application.Calculation = xlAutomatic
    Application.ScreenUpdating = True
End Sub
```

Figure 11. Excel macro deleting workers.

Within the data provided by the company, there is a large number of assemblies made by workers who are not usually on the line, for example, can be new workers, workers who replace another for vacation or illness, or even leaders of each line that are to replace the normal workers in case they need to go to the bathroom or take a break. These workers do not represent the usual behaviour of the factory, so they have not been included in the analysis of their work times. To be able to discard them more easily among the thousands of workers of the company, this program has been created that eliminates all the workers who have made less than 20 assemblies in each line. To do this, two new columns have been created; the first is responsible for counting all the assemblies of each worker and the second eliminates all workers who have less than 20 tasks performed in a station.
Finally, due to the high number of workers and stations available, the decision was made to program another code that ordered the workers in three columns according to the group to which they belonged.

```
Sub ClassifyWorkers()
'
'
'
    Application.ScreenUpdating = False
    Application.Calculation = xlManual
    i = 2
    Lrow = Range("R7").Value
    Range("Q7").Select
    ActiveCell.FormulaR1C1 = "CountWorkers"
    Range("Q8").Select
    ActiveCell.FormulaR1C1 = "Fast"
    Range("Q9").Select
    ActiveCell.FormulaR1C1 = "Normal"
```

```
    Range("Q10").Select
    ActiveCell.FormulaR1C1 = "Slow"
    Range("R7").Select
    ActiveCell.FormulaR1C1 = "=COUNT(C[2])"
    Range("R8").Select
    ActiveCell.FormulaR1C1 = "=0.25*R[-1]C"
    Range("R9").Select
    ActiveCell.FormulaR1C1 = "=0.5*R[-2]C+R[-1]C"
    Range("R10").Select
    ActiveCell.FormulaR1C1 = "=0.25*R[-3]C+R[-1]C"
    Range("U6").Select
    ActiveWorkbook.Worksheets("S120").Sort.SortFields.Clear
    ActiveWorkbook.Worksheets("S120").Sort.SortFields.Add Key:=Range("U6"), _
        SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal
    With ActiveWorkbook.Worksheets("S120").Sort
        .SetRange Range("T2:X74")
        .Header = xlNo
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
    Range("Z1").Select
    ActiveCell.FormulaR1C1 = "Fast"
    Range("AA1").Select
    ActiveCell.FormulaR1C1 = "Normal"
    Range("AB1").Select
    ActiveCell.FormulaR1C1 = "Slow"
    Range("Z2").Select

    For i = 2 To Lrow
    If i < Range("R8").Value Then
    Cells(i, 26).Value = Cells(i, 24).Value
    End If
    If i > Range("R8").Value And i < Range("R9").Value Then
    Cells(i, 27).Value = Cells(i, 24).Value
    End If
    If i > Range("R9").Value And i < Range("R10").Value Then
    Cells(i, 28).Value = Cells(i, 24).Value
    End If
    Next i
    Application.Calculation = xlAutomatic
```

```
    Application.ScreenUpdating = True

End Sub
```

Figure 12. Excel macro group of different workers.

The program in figure 12 is the most important and is responsible for ordering workers in three different groups. In order to carry out this task, the first thing that the program must know is the number of workers that is in each station, because it has to distribute all the groups in an equitable way and not all stations have the same number of workers. To be able to do it, cells will be created dedicated to the counting of workers of each station, then depending on the total number of workers of each station they will be divided into 25% fast workers, 50% of normal workers and a remaining 25% of slow workers. With all this data obtained the program will order in Excel the group of workers with the longest time at the lowest processing time.
Once these data are ordered, a loop is responsible for going through each cell and placing each worker in three different columns, depending on whether they are fast, normal or slow.

5.3.2. Stations Failures

The module leader in addition to providing data on the processing times of each station also offered data on the losses and failures of each work centre. In this way, the availability and average repair time (MTTR) of each problem in Plant Simulation could be introduced, simulating with total credibility the real behaviour of the line. The following table 3 shows what raw data looks like:

Loss Date	Shift	Time	Category	Subcategory	reference1	reference2	Classification	Duration
2019-01-07	Day	11:16	Availability	Equipment Failure	Överordnat system	Others	Equipment	48,00
2019-01-07	Day	11:12	Quality	Scrap and Rework			Operator	46,00
2019-01-07	Night	00:14	Quality	Scrap and Rework			Operator	41,00
2019-01-07	Day	11:14	Availability	Equipment Failure	Dragare/Dragarburk	Others	Quality	24,00
2019-01-07	Evening	19:27	Quality	Scrap and Rework			Quality	50,00
2019-01-07	Day	13:55	Availability	Setup and Adjustment			Material Handling	30,00
2019-01-07	Night	23:28	Performance	Reduced Speed			Material Handling	43,00
2019-01-07	Day	13:33	Availability	Setup and Adjustment			Material Handling	34,00
2019-01-07	Night	03:55	Performance	Reduced Speed			Quality	37,00
2019-01-07	Evening	22:01	Performance	Reduced Speed			Quality	36,00
2019-01-07	Day	07:53	Availability	Equipment Failure	Dragare/Dragarburk	Component out of lin	Material Handling	21,00
2019-01-07	Day	11:07	Availability	Setup and Adjustment			Material Handling	26,00
2019-01-07	Evening	20:31	Quality	Scrap and Rework			Operator	47,00
2019-01-07	Day	09:33	Performance	Reduced Speed			Operator	45,00
2019-01-07	Day	09:33	Performance	Reduced Speed			Equipment	36,00

Table 3. Bad times of the production

The data shown has been manipulated and modified in order to keep the company's confidentiality.

As in the previous table, this table cannot be entered directly as input data in Plant Simulation, so the data has been divided into the three corresponding teams and a new column has been generated with a new error classification system that has been proposed to the company, since the previous system led to confusion and in this way the failures can be studied individually and observe the reaction of each and the seriousness in the real line without suffering costs. This new classification system is divided into five groups that have been called: AGVs, material handling, equipment, quality, and operator, shown in table 4. With all this data a pivot table has been generated, in which the columns are all the stations of the line, the rows are the new classification that has been made and the interior of the table is the sum of all the times according to the season and the classification that corresponds to.

Row Labels	04S120	04S130	04S140	04S150	04S160	04S170
AGV	0,6925	0,021111111	0,370555556	0,374722	0,1425	
Equipment	3,22472222	0,176944444	0,184166667	0,327222	0,6763889	0,673056
Material Hanc	7,30111111	0,729444444	0,584166667	0,612778	0,0333333	3,816944
Operator	0,93583333	0,641111111	0,378611111	0,695556	0,0341667	0,610278
Quality	0,81694444	0,012222222	0,007777778	0,105278	0,0630556	0,135833

Table 4. Classification of bad times.

The data shown has been manipulated and modified in order to keep the company's confidentiality.

This table contains all the times that the stations have failed in decimal format, although the times will be worked in this format because the use to be made of the table is only to calculate the probability of failures of each station (Availability), so it will not affect the final result. To calculate the availability in addition to the time of failure of each station you need to know all the time that these stations have been in operation, for this it has been taken into account every day that the factory has been working, taking into consideration holidays, working days of the workers, meetings, and breaks of the workers, and even the time of weekly maintenance of the line.
The necessary data is already available to calculate the availability of each station, now it is only needed to determine with a simple formula like E.Q.2 of proportions, the proportion that each station works correctly:

$$Availibity = \frac{Uptime}{TotalTime} * 100 = \frac{TotalTime - FailureTime}{TotalTime} * 100 \quad \text{E.Q.2}$$

Applying this formula to each element of the table, the following table is calculated:

AGV	99,8617902	99,99578663	99,9260442	99,92521	99,97156	
Equipment	99,3564071	99,96468527	99,96324385	99,93469	99,865006	99,86567
Material Hand	98,5428378	99,85441684	99,8834115	99,8777	99,993347	99,23821
Operator	99,8132256	99,87204648	99,92443646	99,86118	99,993181	99,8782
Quality	99,8369535	99,99756068	99,9984477	99,97899	99,987415	99,97289

Table 5. Calculation of bad times.

The data shown has been manipulated and modified in order to keep the company's confidentiality.

As can be appreciated, having divided the general failure rate into different groups, the table 5 obtained are very high meaning that the number of failures in each section is lower.
As the last section in this subchapter, the time corresponding to each failure in each station will be calculated, for it simply has been used the tool "Filter" of Excel and an average of the times of each failure has been obtained according to its classification and station. This duration of the failures then has been converted to decimal format to be used in the Plant Simulation software.

6. Creating Process of the Simulation Model

In this subchapter, the reader is introduced to the compression and creation process that has been carried out during this project, the reader will be able to recreate an identical project that replicates with great truth the attitude of the workers of a production line, merely following the steps mentioned below.

6.4.3. Assumption of Data

Whenever a simulation model is developed, it is impossible to take into account all the real data of the line, either due to the impossibility of obtaining them or because of the waste of time involved depending on the precision of the results.
In this subchapter will be detailed the data that have been obviated for the construction of the model.

- The existence of a team leader, all lines have six stations in which there is an operator working at each station, but there is also another member to direct the line and for the replacement of the operator if necessary, this member is known as team leader and has not been considered in the model because the workers that are represented will always work without the need to leave the line.
- The behaviour of each worker could have been simulated instead of having generated group behaviours.
- The distance between the stations has not been considered either because the seconds elapsed between each station are included in the Takt Time, besides the processing time of each worker also includes this time.

6.4.4. Building of the Digital Model on Plant Simulation

In this subsection, the development of the model in Plant Simulation will be thoroughly explained, following a chronological order according to the execution of each step of the development. This section will not deal with a single model, because the multitude of data obtained has allowed replicating different models to corroborate the accuracy of the results and be able to choose the one that best suits the real line. A total of three final models have been generated:

- Fast, Normal and Slow. The model that uses the curve parameterized as "Processing Time", taking into consideration the classification of workers.
- Combination. This model, like the previous one, uses a parameterized curve as "Processing Time", but it differs from the first in that the curve is for all the workers of the line, without classifying them.
- Empirical In this model a variant has been generated for each "Processing Time", and the variants are produced in a cyclic sequence, replicating the obtained data with total accuracy.

The figure 13 corresponds to the model "Fast, Normal and Slow", this figure has been used because it is the most complete model of the three, so the others only differ in the elimination of certain parameters. At first glance, it can be appreciated that the virtual environment generated does not have any of the tools offered by the software for the simulation of lines with workers. In their place pallets have been taken, which have been programmed to be generated in a random order, but the number of pallets produced by each group of workers will be controlled by a table with the percentages of each of them. Each pallet has a name and, according to this, a predefined "Processing Time" is executed by the formula of the distribution that has best adjusted to its real working times. For the generation of these curves, the Experfit or DataFit software (software integrated into Plant Simulation) has been used, which have generated some histograms, an example of this kind of histograms can be seen on figure 13.

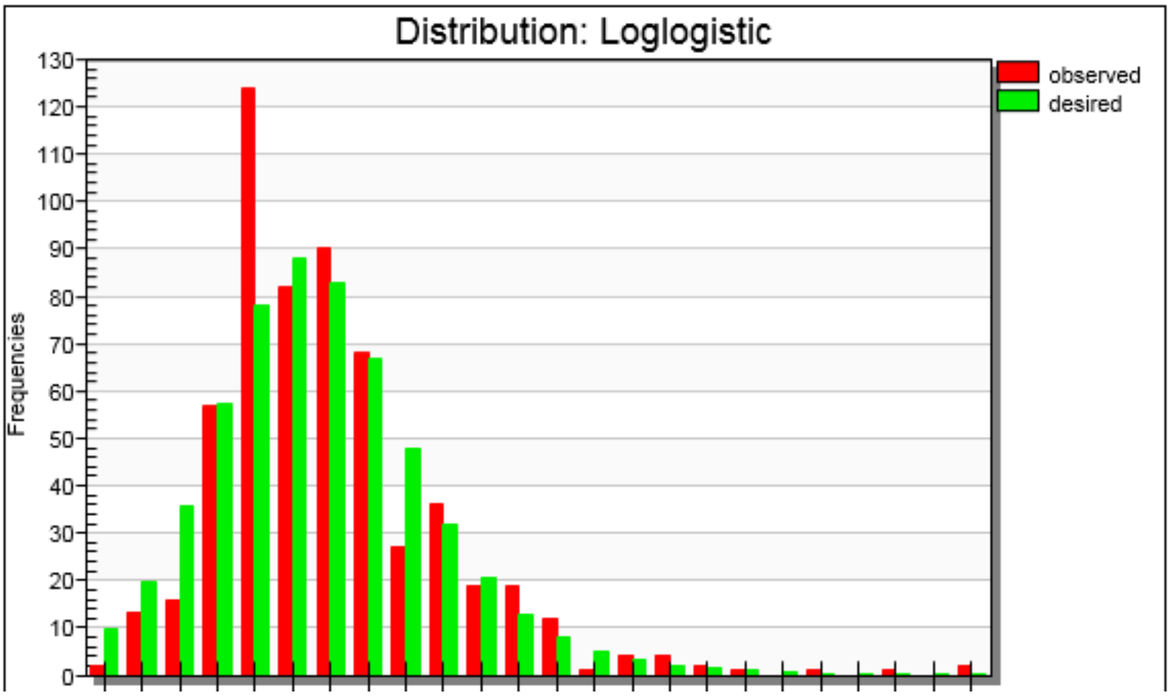
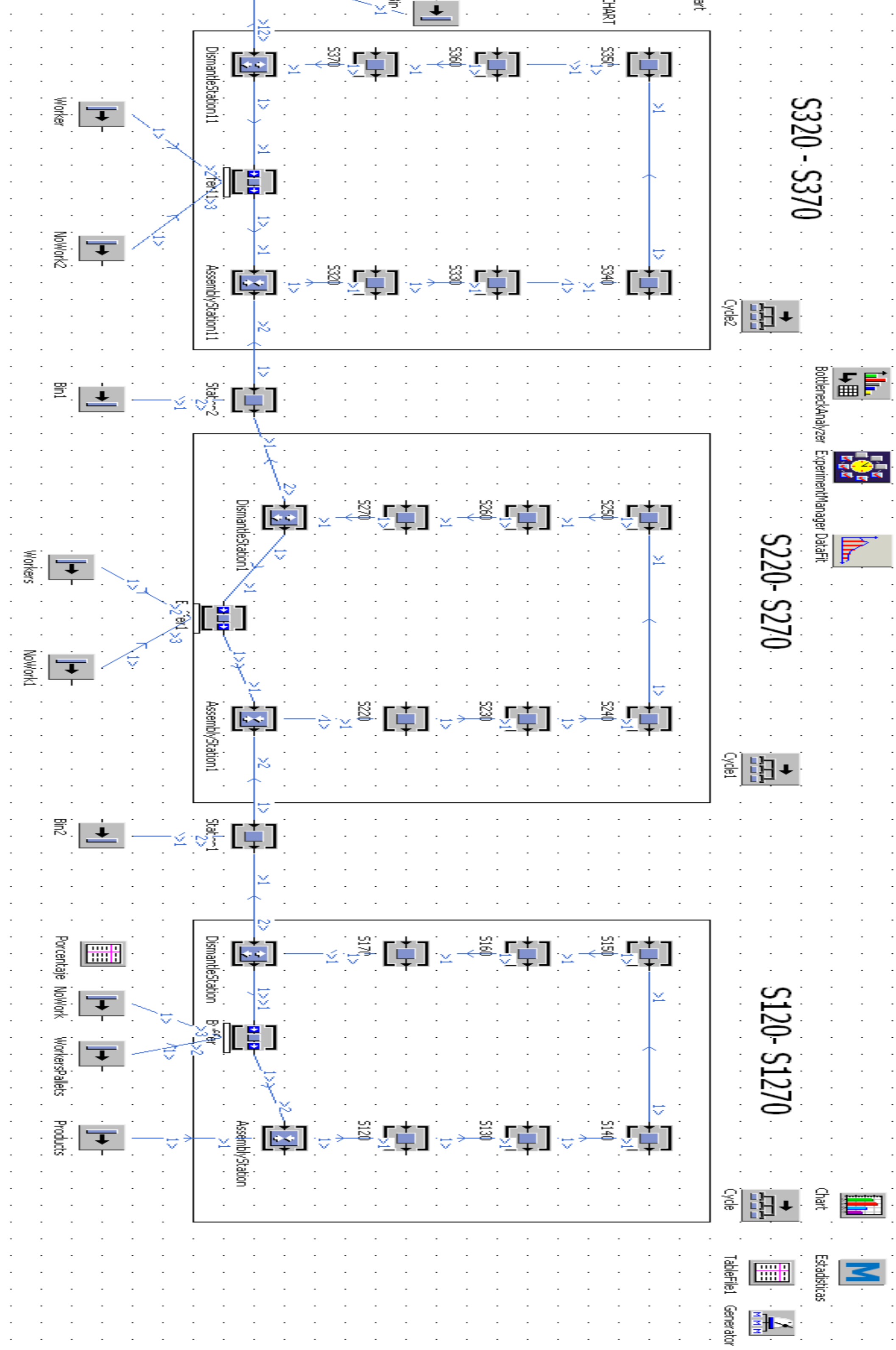


Figure 13. Graph of the model "fast, normal, slow"

with the data entered, and superimposed on them, the chosen distribution curves are adjusted following the criterion of choice, the "Pearson Chi-square". As can be seen in the figure 13. Experfit tries to adjust as much as possible a parametrized graph to the different times, but

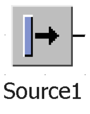


nall differences are

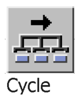
Figure 14. Picture of the model" fast, normal, slow".

As mentioned in the description of the software, Plant Simulation has an object-oriented programming language, these objects are the small squares that are seen in the image, these squares are configured as standard with some attributes that can be modified, and even reprogrammed them and include in the new functions that they can develop. In this section, the attributes and programming of each object will be explained to meet the requirements of our line. Figure 14 is an example how a simulation model could look like with all the objects.

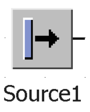
1. WorkersPallets: This object is the Source type, and it has been modified for the production of pallets. Each pallet represents an operator, the pallets are generated in a random way, changing its name according to the percentages assigned to it in the "Percentage" data table. Only the order of introduction of the pallets in the line has been randomly programmed, but the number of pallets of each group is always controlled from the aforementioned data table, which is called "Percentage".



2. Non-Workers: This object like the previous one is of the Source type, and generates pallets as well. These pallets always have the same name, "NotW". The creation of this object has the intention of controlling the number of workers that are on the line and must add eight pallets in total among the workers generated by Non-Workers and WorkersPallets. Since the pallets do not pass through other lines until they reach "Drain", they rotate on the same line. In addition, all stations and assembly stations must always have a pallet since the object "Cycle" prohibits the execution of empty cycles so they must always add eight.



3. Cycle: This object serves to control the transfer of products between stations. It allows generating a Takt-Time, in which the cycle time will be controlled by "Products". "Cycle" prevents that empty cycles can be done, this means that all the stations of each line must have completed their work to be able to advance to the next one. There are three "Cycles" in the model, since buffers and disassembly stations cannot be included in the object, however, these two objects have a processing time of zero, so their influence on the Takt -Time is null since these two mentioned objects must always wait for the first station of each line. As an aside we can add that it can be replaced by a "Generator" object that introduces each cycle time a method that indicates if the stations are ready to transfer the product to the next station, although its programming is more complex besides introducing errors if all stations they are not ready when the respective cycle time has finished.



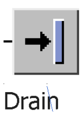
4. Products: Is responsible for generating the parts that will be assembled together with the pallets in the assembly stations. The production number of parts is infinite and will be generated every cycle time, therefore it is responsible for regulating the cycle times of the module. Each part starts with a default name, but it can be modified if it is mounted on a pallet called "NotW".

5. Assembly Stations: Its function is to join workers (pallets) with their respective products. They have a null processing time, so they do not affect the production cycle.

6. Stations: Are the main objects of the module and the only ones that represent something tangible in the industry. Each one has an integrated programming code that is activated when the product is entered into it. When this code is executed, the station varies its processing time cycle by cycle depending on the type of worker that is according to the corresponding parameterized distribution. Although two workers of the same group work in the same station, they will not have the same processing time because, as mentioned, times vary according to a function. In each station, they have also integrated their corresponding faults, divided into five groups and obtained from the tables shown above. The first stations of each line modify the name of the part of the product if it is assembled on a pallet called "NotW", to "Trash". In this way, these parts can be removed from the drain and not affect the final throughput. The stations that are between lines and at the end of the module do not have any processing time and has the sole purpose of sending the products to their corresponding place, eliminating the wrong parts of the final output, hence the Drain always throws the results of the products that have been worked by a valid operator.

7. Dismantle Stations: Divide the pallet of the product. The pallet is sent to the buffer that will send it to the assembly station to be reused, and the product will continue to receive the operations of the successor stations. They have a null processing time and together with the assembly stations act as buffers of unique capacity since they keep the product until the next station has completed its previous work and is ready to continue.

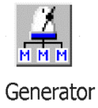
8. Buffers: Are responsible for storing the finished material to be used immediately in the next process, the buffers represent AGVs ready to continue to the successor station. There are two types of buffers in the model, those that receive the pallets to be reused within the same line and those in charge of transporting the products between different lines. Both have a null processing time, but differ in that the first is fictional buffers forced to enter the model, because it is not possible to link an assembly station with a disassembly directly. And the second ones are real buffers with a single capacity (they only have the capacity to store a product because the assembly and disassembly centres act as buffers with a single capacity as well, so in total it gives a buffer with three units of capacity equal to real model). This last type of buffers has not been included in the “Fast, Normal and Slow Model” because the stations between lines are acting like buffers fulfilling the requirement of having space for three AGVs between each line.



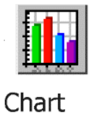
9. Drain: This object is the delimiter of the model, and offers final production statistics, such as the final throughput or the average life of a product on the line. This object will be very useful in the next chapter of results analysis. There are two objects of this type, the first called "Drain" in which all valid products are received, and the second

10. called "Bin" in which only the products that have been assembled with a pallet that represent the lack of a worker.

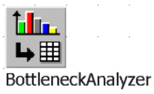
11. Method: Allows to modify the characteristics of an object, or to program an action. Each station can carry an integrated method without the need to use this object, in the model it has been used as an exit action in the last station to address the pallets and also introduced in the generator to generate production statistics.



12. Generator: Allows to reproduce a method every certain time interval. In our case, it uses a method to record the statistics of the line every half hour, in a data table.



13. Charts: Allow to draw the data collected through the data tables. It has been used to obtain the stability graph of the system.



14. Bottleneck Analyzer: Tool integrated into Plant Simulation for the detection of bottlenecks.

15. Experiment Manager: Tool that allows the realization of multiple simulations and offers detailed statistics of the parameters that we select. This tool is responsible for selecting parameters such as the final throughput along with its standard deviation to determine the number of replications needed to obtain a certain confidence interval.

With the explanation of each object and the use that has been given to them for the construction of the model, the operation of the line can be understood perfectly. In the following paragraphs will be explained the logic that has needed to be programmed by simple diagrams and will be exposed to the differences between the different models.

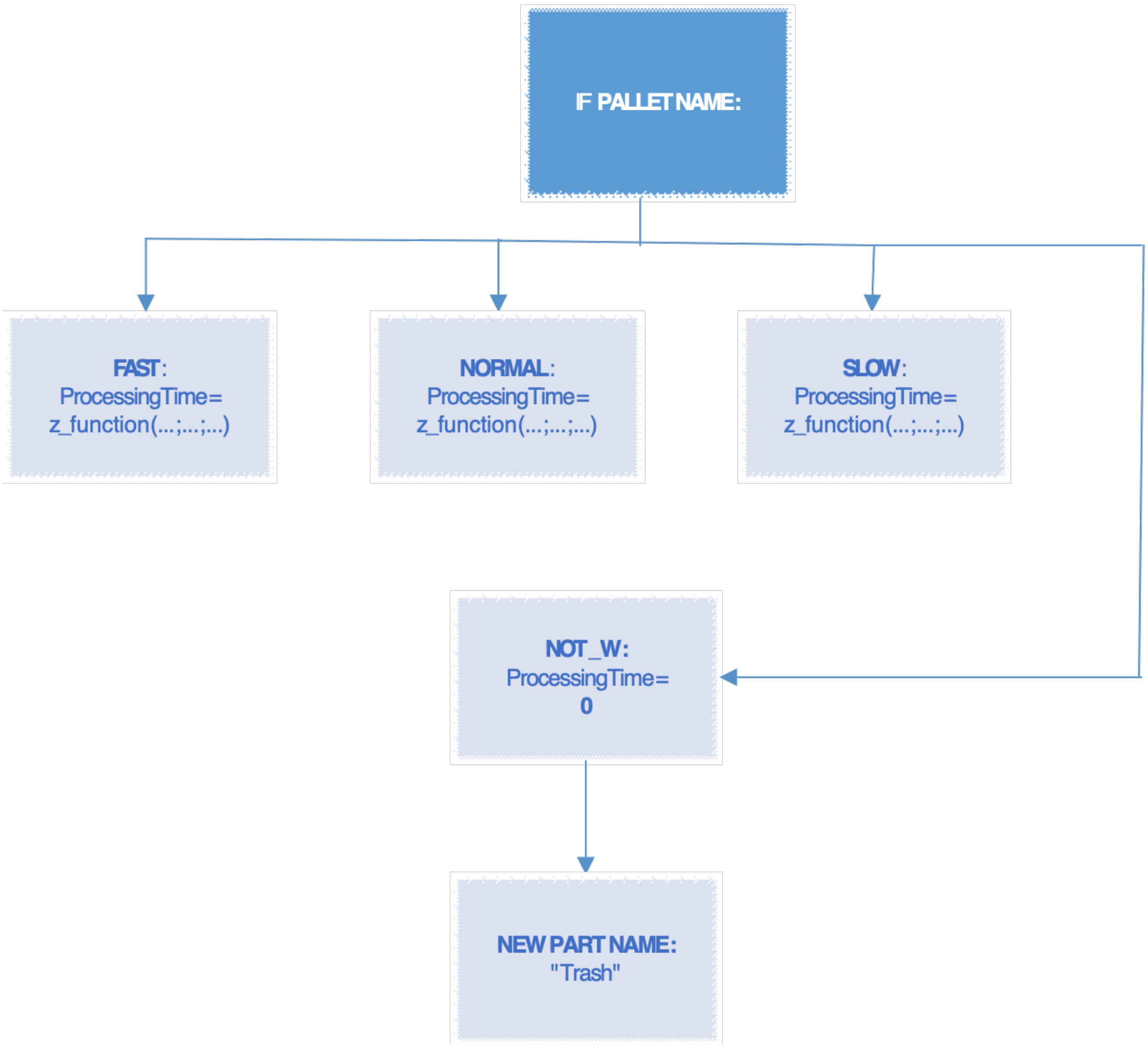


Figure 15. Logical diagram integrated into each station.

The diagram presented, in figure 15, describes how each station modify his processing time taking in consideration the type of pallet (worker); that arrives to the station. This logic is an entry action, that means that the logic is executed at the moment the pallet arrives to the station. If a pallet called "Not_W" arrives to a station, the station will rename the engine as "Trash" and all the stations of the line will not work on that part.

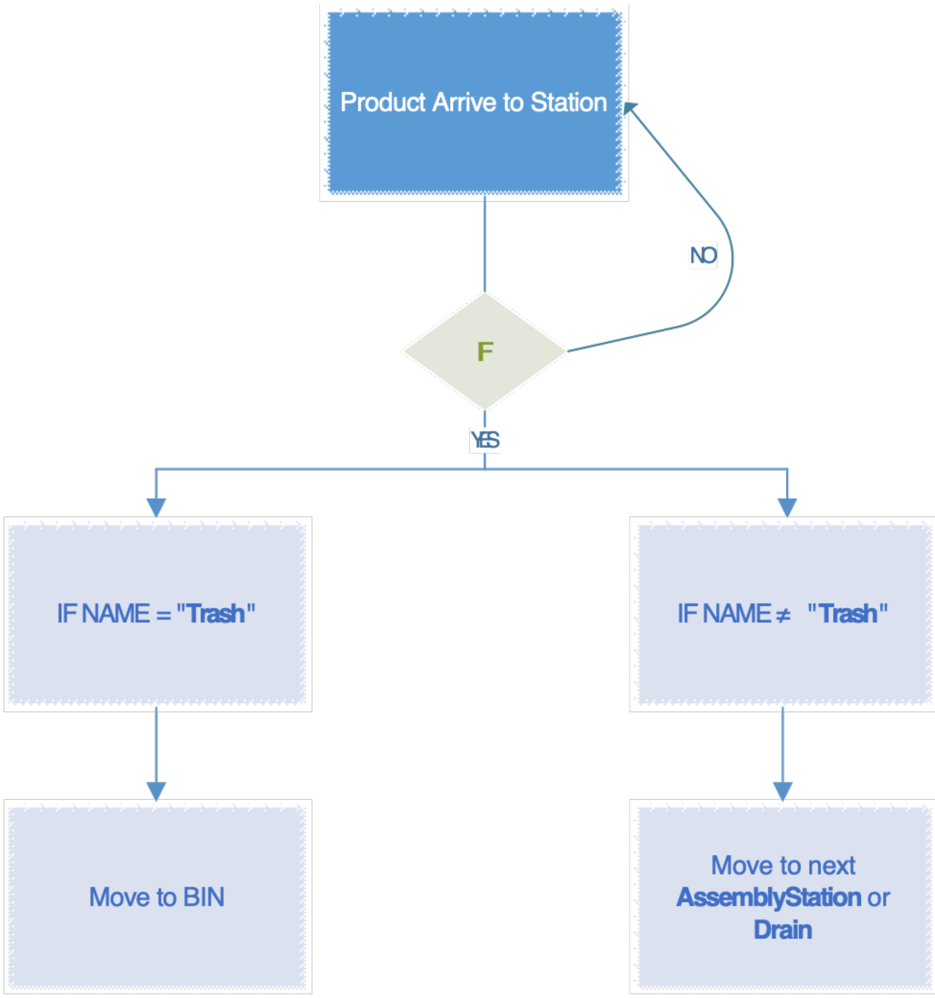


Figure 16. Logical diagram of stations dedicated to the shipment to the correct location of the parts.

The diagram shown in figure 16 defines how the pallets are removed from the line. This logic is only executed in the last stations of each of the 3 lines. To do not affect the rest of the lines. If a engine called “Trash” (this types of engines represent the lack of a worker in the line) arrives to a station, that station will have a processing time of 0 and will act like a buffer.

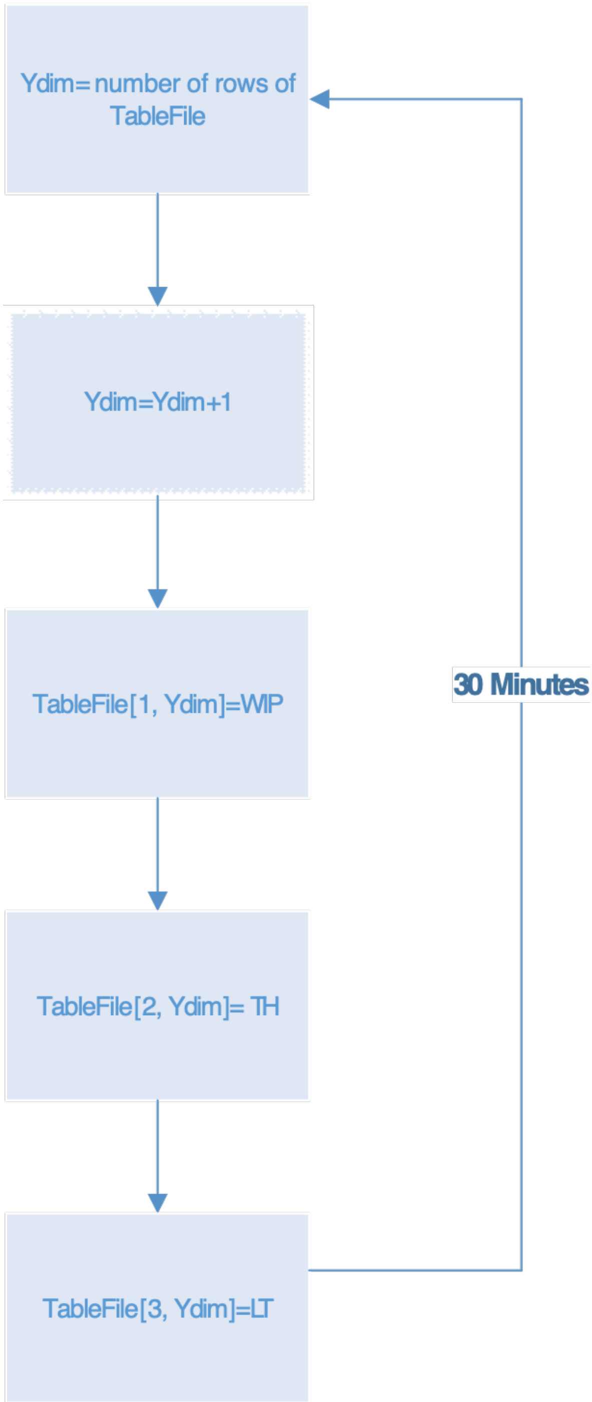


Figure 17.Logical diagram of generator programming by a method for the collection of line statistical data.

The figure 17 is presented, just represent how the information has been collected, each data collected has to be inserted in a cell, the diagram represent which data should be introduced in each column and row. The first column is for the work in process, the second one is for the throughput, and the third one represents the lead time. All this information is collected each thirty minutes, by an object called “Generator”.

6.4.5. Model without worker classification

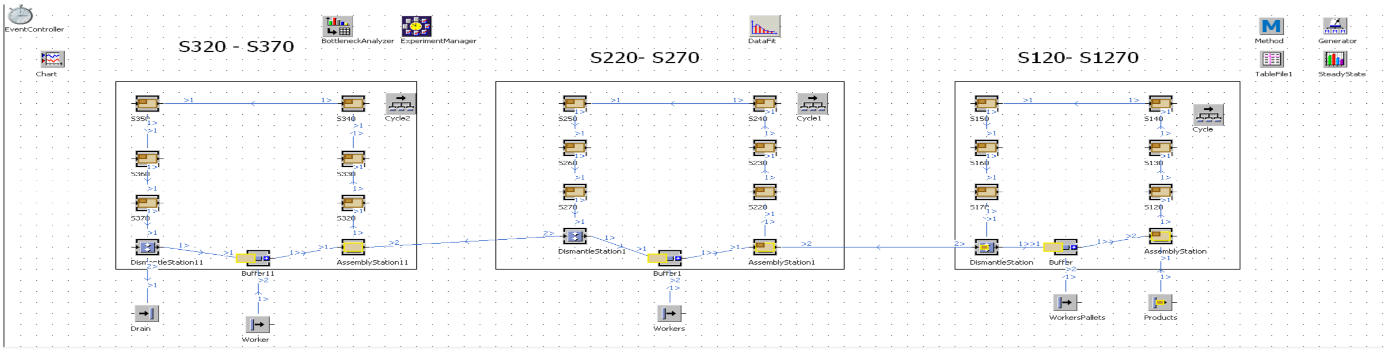


Figure 18. Picture of the model without worker classification

This model that is shown in figure 18 has been constructed with the sole purpose of observing the benefits of classifying workers in groups. As can be easily observed, it is a simpler model than the previous one, counting on a smaller number of objects than the first. All the pallet producing "Sources" that represented operators absent from the line have been eliminated, along with them the table of data that contained the different groups and the percentage of

workers corresponding to each of them has been eliminated. The last station that was used to choose the final location of the pallets was also eliminated, since now we only have one outlet for the products.

As expected both models must comply with the real behaviour of the line, therefore, the results in the final throughput are very similar, having a difference in the production of only 0.14%. However, the fact that similar results are actually obtained does not imply that the construction of another model was a waste of time, but corroborates that the results are reliable and do not move away from reality even though objects have been added and a logic that allows to control and change aspects of production. In addition, the most important detail of the first model is the possibility of modifying the controlling board of the type of workers that is on the line, but this will be discussed in the following chapters. In figure 19, the difference in throughput/per hour is shown between models ”FNS” and ”Combo”

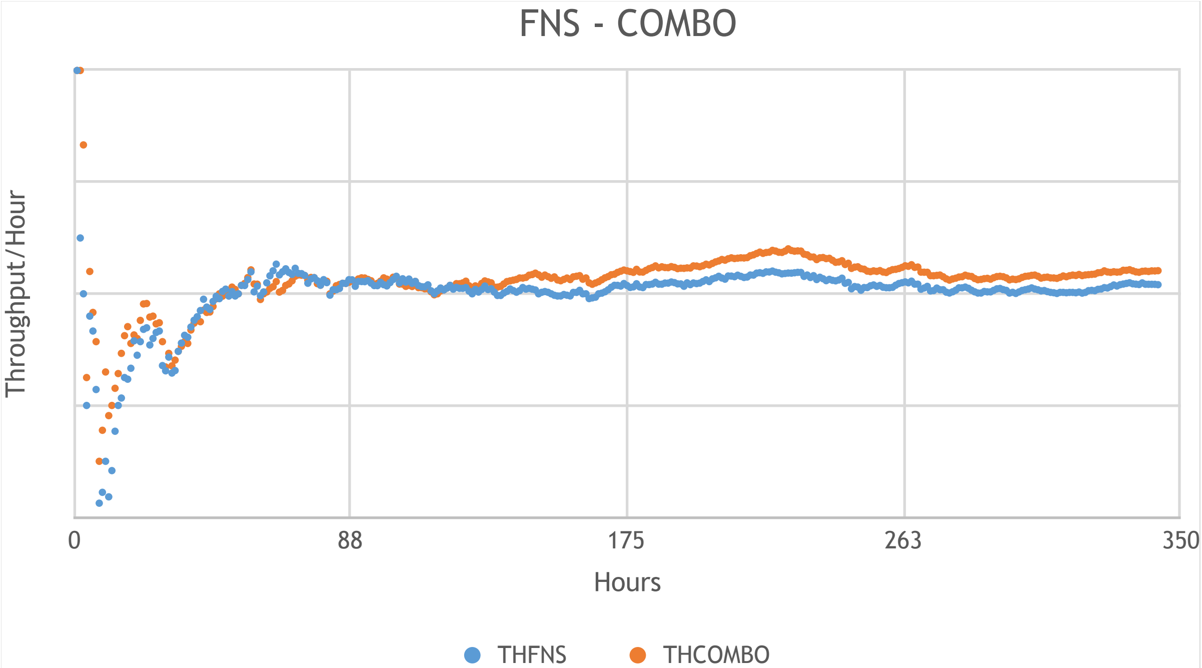


Figure 19. Diagram of models "FNS" and "Combo".

6.4.6. Empirical Model

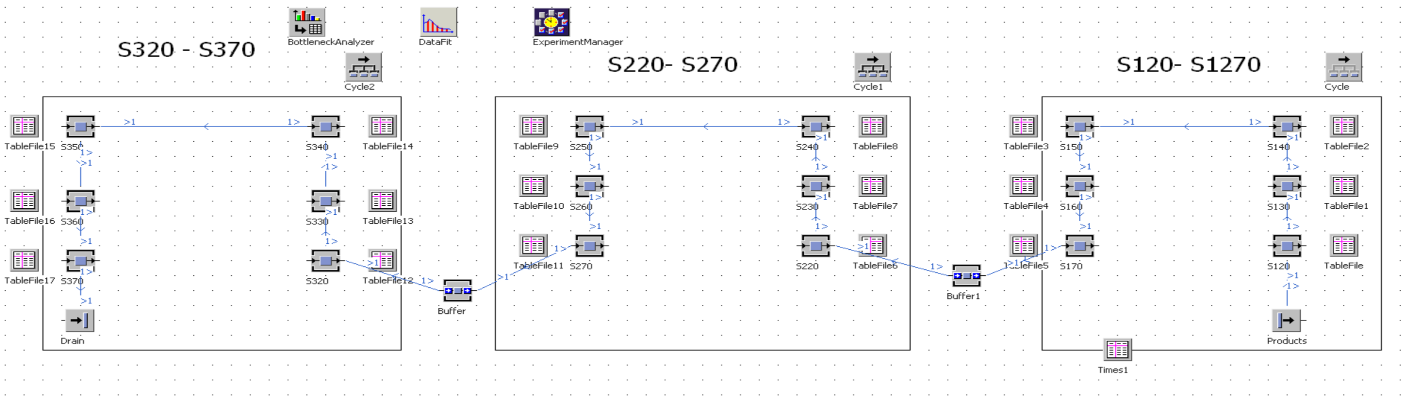


Figure 20. Empirical model.

This model in figure 20 is the simplest of all mentioned, but it has the special importance that it is a model that replicates with great similarity the behaviour of the line. Working with empirical models is not useful for the future, since it does not allow modifying the type of workers or any attribute of the line, nor introduces changes in which the reaction in the final throughput can be appreciated. But they serve as a reference to simulate a model that can be optimized by adjusting to the real performance of the factory.
Due to the high amount of data that had and the dispersion of them caused by the differences of time between the workers of each station, a valid model could not be generated using the tool that gives Plant Simulation for the creation of empirical models. The introduction of the

data in the software is restricted, so the raw data had to be converted to frequency tables where more than seven thousand results per station had to be located according to the percentage of the number of times each processing time appeared. The percentages were so low, that to obtain a true representation more than two decimal places were needed, an option that the software makes impossible when working with the percentages format. So, for the creation of the empirical model, each operation performed was represented by a variant, obtaining then as it has been cited before more than seven thousand variants that must be reproduced consecutively, thus generating a model that replicates with total accuracy the real line.

Like the previous models, there is a great similarity in the final production yield, with the difference between both models of 2.17% as it shown in figure 21. However, the difference is greater than between the first models because the processing time that is executed in each action is completely real and random, whereas in the previous prototypes each processing time follows a distribution that tries to adjust as much as possible to the results but it is still an approximation.

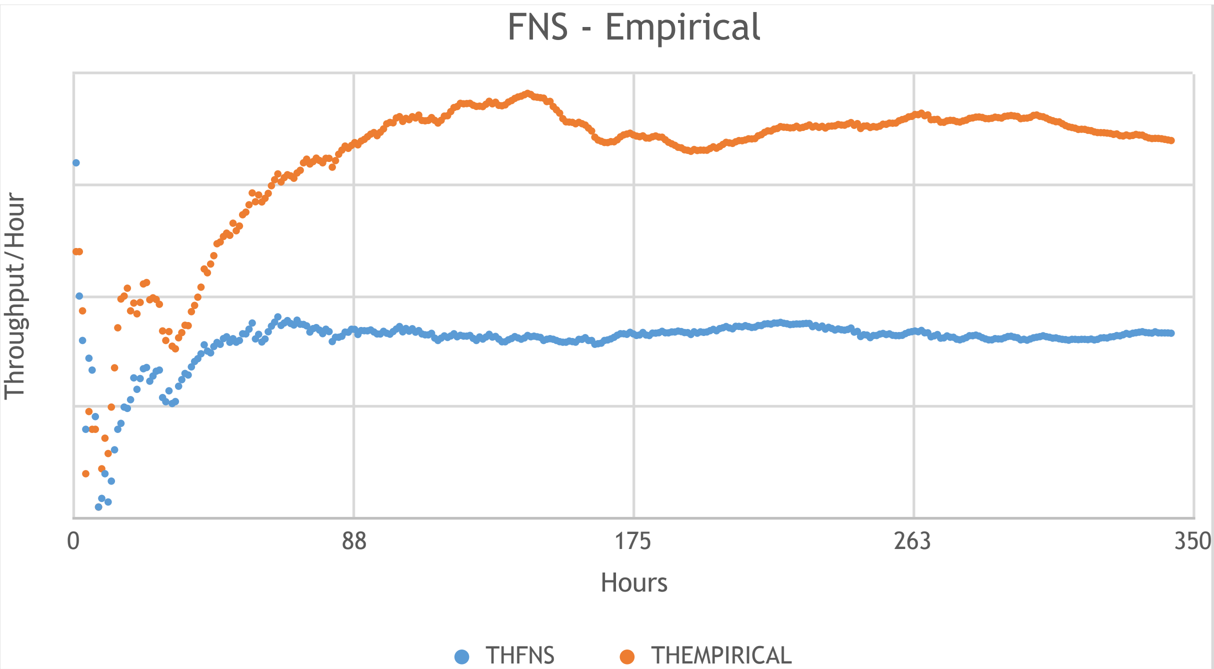


Figure 21. Diagram of models" FNS" and "Empirical"

7. Validation

According to the methodology explained, checks of the model must be performed during its construction and especially once finished.

The verification process has been carried out with the production engineer in charge of managing the production flow of the line in the company. In this meeting, a presentation was made where we could show him the functioning of the three models, how they were built and the parameters we have taken into account. In addition, the results of the model could be verified and verified that they were the expected results.

The process took into consideration two aspects: compare the results of the simulation model with the historical data provided by the company and by the Little's Law.

The results of the model such as TH, WIP or lead times shown in table 6, were compared with data collected by the automated system owned by the factory. All these data were exactly the same results that the model can show. The detection of the bottlenecks in the stations causing the blockage in the line was the same predicted by the simulation. The considered results of the TH of the digital model behaved exactly like those of the real line, to such an extent that the results of eliminating one worker in each of the three lines were still identical if that number of workers in the factory were eliminated. Also, the great importance of a slow worker on the line could be proven, affecting a single worker to the complete module, since all the stations and the system of AGVs are synchronized among them.

% of Simulation	Real TH	Real LT	Calculated WIP	RealWIP	Difference (%)
10	62.39745	0,5026	31.36005	31.26585	0,3015
20	62.63895	0,4996	31.29555	31.2565	0,2238
30	62.5605	0,5001	31.2855	31.0647	0,7111
40	62.54985	0,4990	31.21065	30.9549	0,8262
50	62.679	0,4972	31.1634	30.85605	0,9960
60	62.77695	0,4959	31.1325	31.1451	0,0405
70	62.6565	0,4971	31.1454	31.26585	0,3852
80	62.5752	0,4981	31.17	31.26585	0,3068
90	62.6172	0,4979	31.17915	31.3353	0,4985

Table 6. Results of TH, WIP and lead times

The data shown has been manipulated and modified in order to keep the company's confidentiality.

The following figure 22 shows the linear correlation between the LT and the TH. As can be seen at lower LT, the greater the TH. This difference is obvious the less time the product spends on the line, the faster the production time will be, and the higher production rates can be reached. It is also true that the WIP does not remain constant during the whole simulation, but its decrease is minimal in comparison with the difference of the attitudes in the workers, which allow such different times in the LT. In addition, if the number of workers is altered, leaving certain stations empty, the WIP would remain constant because the empty stations would act as buffers of the line.

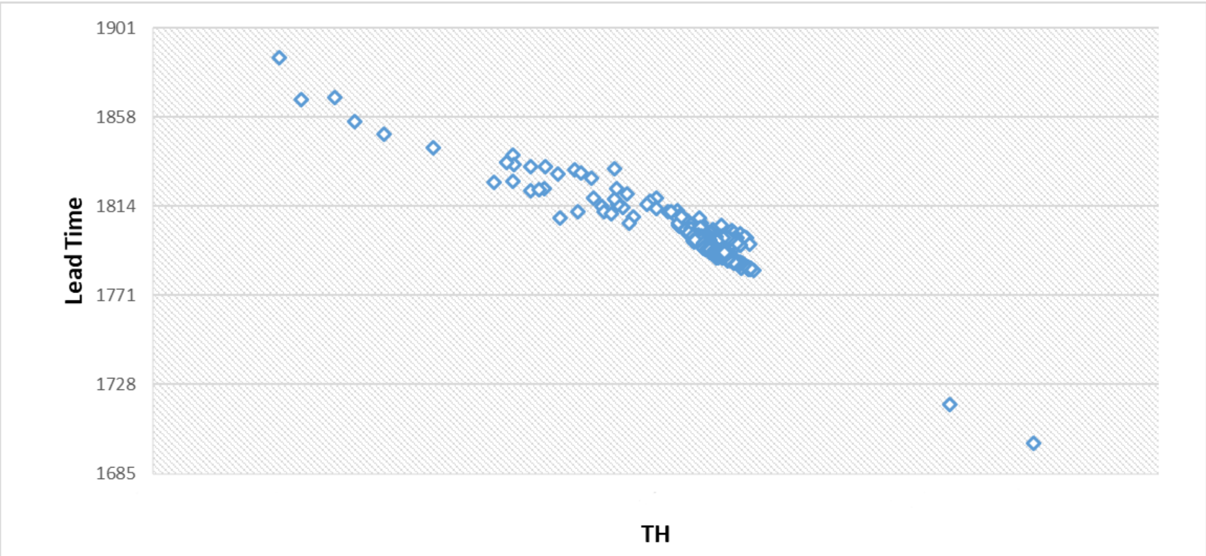


Figure 22. Linear correlation.

8. Results and analysis

This chapter presents how the model has been validated, the analysis that has been executed and the elimination of factors that can alter our TH or add variability to the result. Within the variability obtained, confidence intervals will be studied, to know what is the average of the results and if they are apt to be considered standard or deviations from the real result.

As it has been explained, some analyzes are needed to study the inherent variability of the project and to be considered as finished. In this chapter, two types of analysis will be calculated: warm-up time analysis and replication analysis. On the one hand, the warm-up analysis determines the time it takes for the model to reach stability. The model can be considered stable when its response is no longer transitional and a constant result can be seen in the output of its products. Since the model reaches a stable state, it is possible to start measuring the statistical analyzes. In this section, the run-length analysis could also be cited, but the final result is not stochastic and is determined without almost any variability. Normally for non-terminating simulation models, the Welch method must be performed to determine the warm-up time, and Robinson's convergence run-length method to determine the necessary simulation time. But, as can be seen in the figure 23, where is compared the throughput per hour of the module with the time that the line is working, the built model reaches absolute stability in a period of 16 hours, being able to consider a period of 15 days, as a considerable time to obtain a steady-state sample of the simulation.

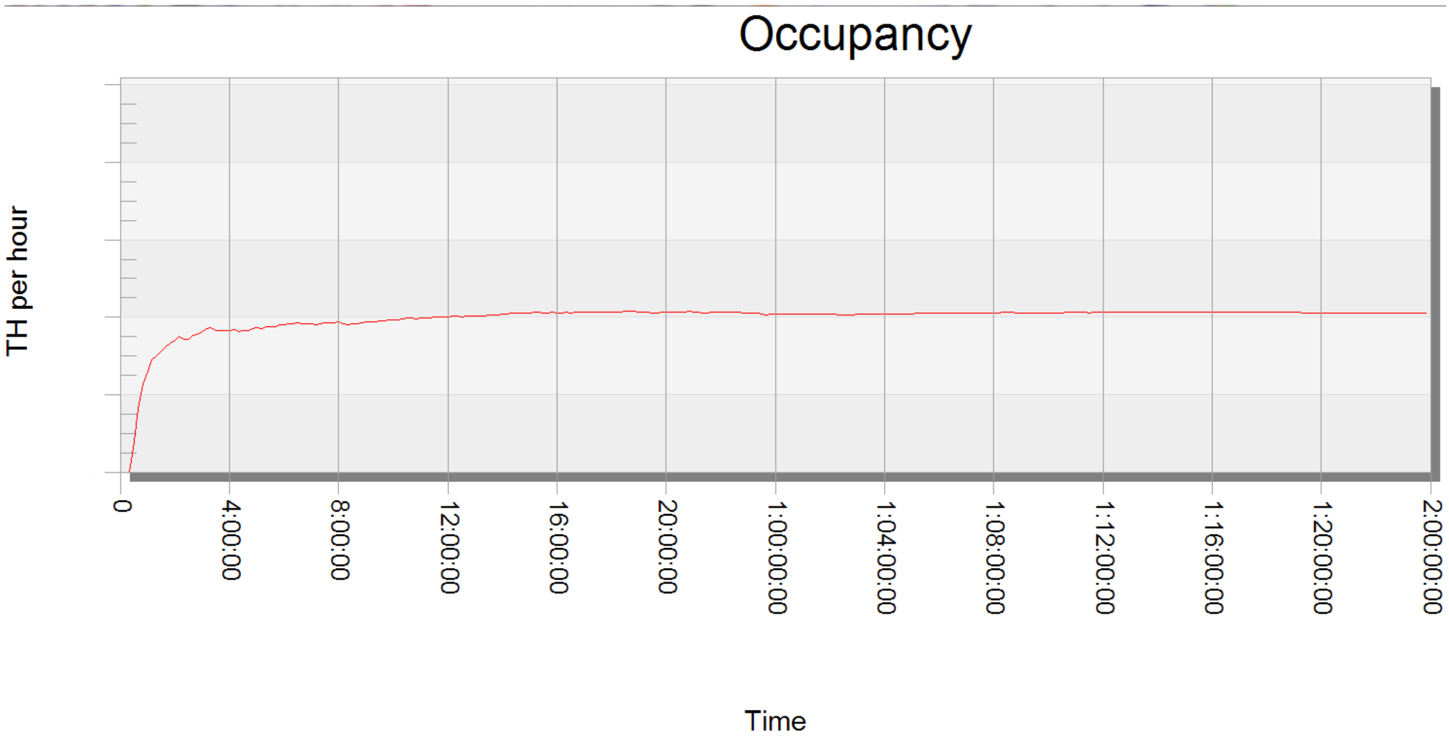


Figure 23. Occupancy

On the other hand, replication analysis is the analysis responsible for determining the number of replications needed for each experiment. The steps for conducting the analysis are the following:

1. Determination of the data we need in the model.
2. Execution of an initial simulation with "Experiment Manager" with a total of 15 replications per experiment.
3. Find absolute precision using an accuracy of 1%, like E.Q.3.

$$AbsolutePrecision = RelativePrecision*MeanValue \quad \text{E.Q.3}$$

4. Observe the number of replications obtained with the formula TINV () shown in E.Q.4. Which calculates the inverse of two-tailed Student's T Distribution, which is a continuous probabilistic function used to test hypotheses in small data samples.

$$numberofreplications = \left(\frac{\left(TINV\left((1 - CI, (n-1)\right)\right)*stdev}{ap} \right)^2$$

E.Q.4

Where, CI = confidence Interval, n = number of replications, stdev = standard deviation and ap = absolute precision.

5. Check if the number of necessary replications is exceeded and adjust the relative precision, repeating the process as many times as necessary.

The values and results obtained are those shown in figure 24 and figure 25. For a confidence interval of 0.9 and a relative precision of 1% the analysis reveals that there are not enough simulations for the standard deviation of the WIP.

Output	Confidence Interval	Number of replications	MEAN	STDEV	Standard error (Confidence Interval)	Absolute precision	Number of simulations needed
TH	0,9	15	62,421	0,8227	0,3894	0,4160	13,14
LT	0,9	15	0,5071	0,0051	0,0024	0,0051	3,36
WIP	0,9	15	31.0995	0,9612	0,4549	0,2070	72,45

Figure 24. Results for standard deviation.

Output	*Confidence Interval*	*Number of replications*	*MEAN*	*STDEV*	*Standard error (Confidence Interval)*	*Absolute precision*	*Number of simulations needed*
TH	*0,9*	*15*	*62.421*	*0,8227*	*0,3894*	*0,4160*	*13,14*
LT	*0,9*	*15*	*0,5071*	*0,0051*	*0,0024*	*0,0051*	*3,36*
WIP	*0,9*	*15*	*31.0995*	*0,9612*	*0,4549*	*0,4147*	*18,05*
TH	*0,9*	*19*	*62.4558*	*0,7767*	*0,3266*	*0,416*	*13,14*
LT	*0,9*	*19*	*0,5066*	*0,0092*	*0,0039*	*0,0051*	*3,36*
WIP	*0,9*	*19*	*31.0263*	*0,9459*	*0,3978*	*0,4137*	*17,57*

Figure 25. Results of standard deviation.

The data shown has been manipulated and modified in order to keep the company's confidentiality.

Due to the high number of simulations required, the accuracy of the WIP has been raised to 2%. With this elevation we have managed to reduce the number of simulations to a total of 18.05; since this figure is not possible, it is rounded in excess of 19 to increase the accuracy.

8.1 Bottleneck Analysis

In this small section the weak points of the production line will be exposed, first of all the reader will be shown an image that compares the performance of all the stations of the line, allowing to locate the problem.

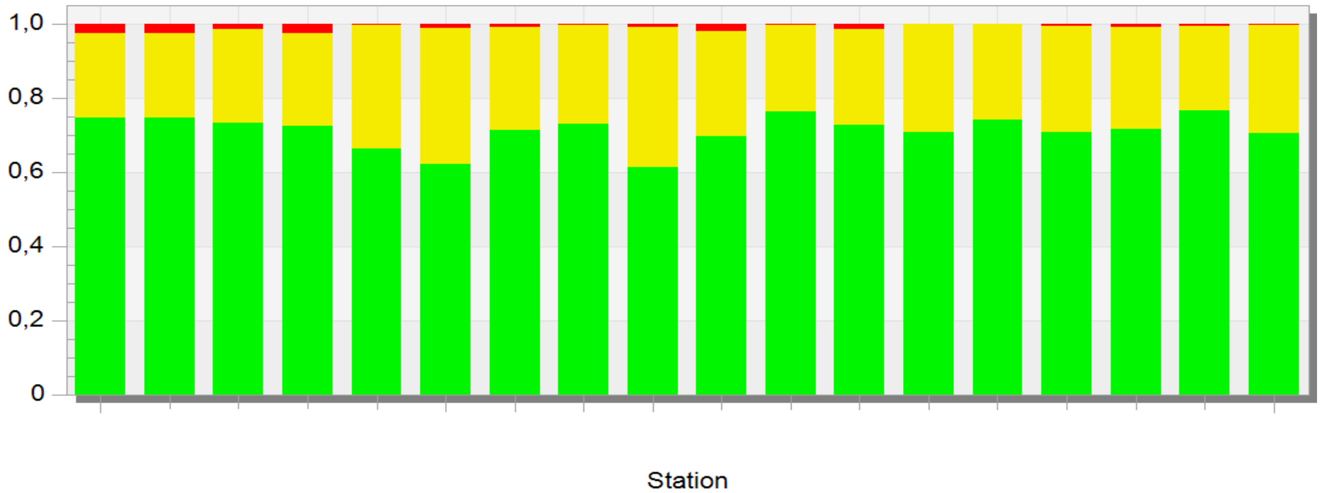


Figure 26. Bottleneck analysis.

In the Figure 26, the green color represents the time the station is working, the red color represents the time that a fault has occurred in it, and for the last one, the yellow time shows the time that the station has been blocked, normally this blocking time reflects the time that the station has been waiting for the successor stations, but it must be remembered that the represented line has a Takt-Time that forces all the stations that are in the same group to wait for the completion of work to be able to advance to the next job. And this is where the problem lies, if the reader takes a look at the figure 26 again, he will be able to see how all the stations are normally balanced. Therefore, no station supposes a significant greater load of work that the others, but it is a great difference between the diverse behaviors of the operators. Therefore, all operators of the line must always wait for the operator who works the slowest, resulting in a whole work group wasting their time due to the operator that is least efficient. Therefore, in this type of lines, which have a Takt-Time, special emphasis should be placed on the training of operators. In addition, as a contribution of the author, it is proposed the classification of workers in shifts of work taking into consideration the efficiency and performance of workers.

8.2 Identification of Takt-Time problems and their solution

As mentioned in the previous sub-chapter, all the stations of the production line find a similar balance between them, so the main problem does not lie in the time difference between the stations. The main problem that the authors detect in the production line of this company, is

the difference in the efficiency of the different workers, this is because the company on which the project is done, follows the principles of the Japanese philosophy Lean. According to these bases, the line must work through work cycles called "Takt-Time", as previously mentioned. This means that no worker of the whole line should continue with the next task until all the workers of the line have finished the task of one cycle. Once all workers have finished they move to the next station to perform the new task and complete the next cycle. To demonstrate that the main problem lies in this line characteristic, the following experiment has been carried out.
Thanks to the characteristic that has been implemented to modify the types of behaviour of workers according to percentages; a simulation model will be executed only with fast workers and the results are shown in the figure 27, again another experiment will be executed only with slow workers, and the results of the production rate are shown in figure 28.

Working:	89.32%	Average lifespan:	28:14.0351
Setting-up:	0.00%	Average exit interval:	1:14.9331
Waiting:	10.68%	Total throughput:	16527
Stopped:	0.00%	Throughput per minute:	0.80
Failed:	0.00%	Throughput per hour:	72.06
Paused:	0.00%	Throughput per day:	1153.05

Figure 27. Fast Workers.

Working:	87.07%	Average lifespan:	2:36:18.7979
Setting-up:	0.00%	Average exit interval:	6:48.1790
Waiting:	12.93%	Total throughput:	3034
Stopped:	0.00%	Throughput per minute:	0.15
Failed:	0.00%	Throughput per hour:	13.26
Paused:	0.00%	Throughput per day:	211.67

Figure 28 Slow Workers.

As can be seen in figures shown above, the difference between the production rates according to the behaviour of the workers is enormous. Obtaining the experiment in figure 27 is a throughput of 72.06 and the experiment in figure 28 is a throughput of 13.26.
But evidently the production line does not work only with good or bad workers, but there is a homogeneous mixture of them, the reader can expect that the good workers will compensate the bad workers and obtain an average between their times. This thought is completely wrong due to the "Takt-Time" characteristic, mentioned above. The rate of production of the line is completely limited to the slowest workers and to demonstrate these results, a third experiment has been carried out where the fast and slow workers will be mixed in two identical halves, randomly distributed throughout the line. The production rate obtained according to this mixture is as follows:

Working:	87.17%	Average lifespan:	1:49:15.6351
Setting-up:	0.00%	Average exit interval:	4:45.5777
Waiting:	12.83%	Total throughput:	4336
Stopped:	0.00%	Throughput per minute:	0.21
Failed:	0.00%	Throughput per hour:	18.9
Paused:	0.00%	Throughput per day:	302.51

Figure 29. 50% Slow Workers and 50% Fast Workers.

The data shown has been manipulated and modified in order to keep the company's confidentiality.

As can be seen in the figure 29 the rate of production of the mixture in two halves of the types of workers is very similar to the production rate of the experiment running only with slow workers shown in figure 28.

These results are surprising due to the amount of paid hours wasted. Therefore, different solutions are proposed, the first of the solutions involve training the workers better so that everyone works in a productive and equitable way. And the second is to unify and put the workers in shifts so that workers with similar efficiency are gathered in the same shift, avoiding that they have to wait a long time for the last worker to finish.

It is understood that there are different attitudes, and workers who work in a slower way are not criticized, the author only proposes to relocate them so they can work in a more comfortable way according to their abilities.

Finally, a graph, figure 30, has been made which classifies the average times of each worker in a line composed of six stations. This graph has been made with the objective of study for classification between the shifts of the operators. If a classification is not made of the workers as it has been observed in the results, all the workers would be limited by the worker who can be seen more to the right, wasting the expertise of the other workers. It is clear that despite the fact that the slowest operators must be trained, the classification for efficiencies in shifts is mandatory.

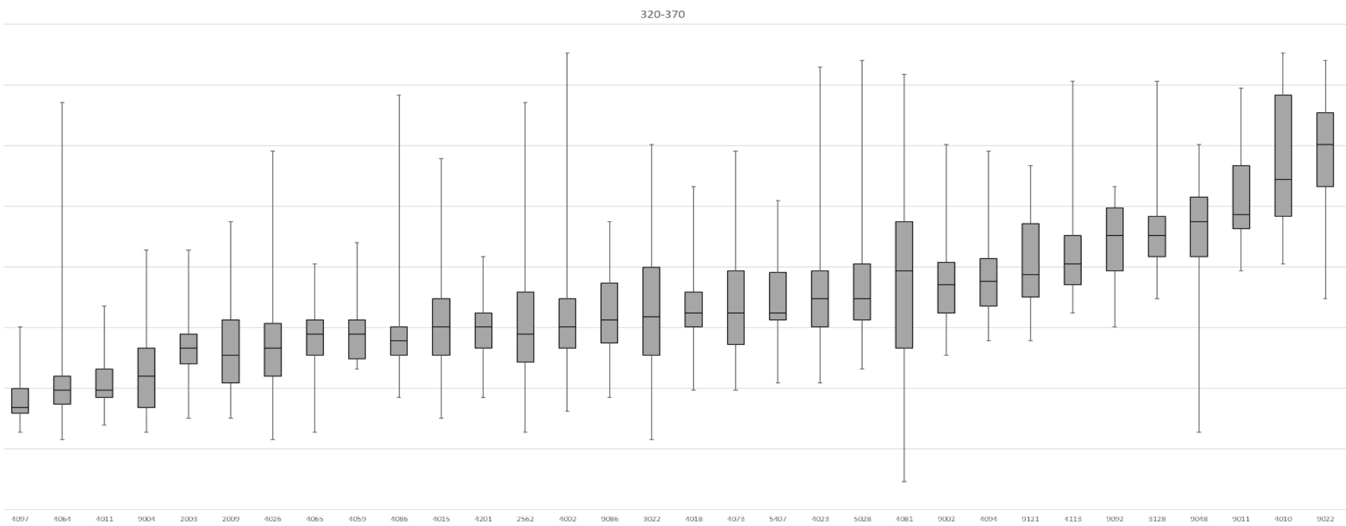


Figure 30. Average times of each worker.

9. Discussion

This chapter will authors discuss the whole project and its progress and the problems that came back and forth. Future work and reflections will also be discussed from the author's point of view.

Data

From the beginning, the data that was collected from the production was only about a few days even if it was big data. For the fact, the more data that are collected the more distribution data will be better and it will show reliable results. A question that came up was how could a few days long data determine how the assembly line works. At a later stage, the data that were ready worked to be an input data was showing poor results and it would be unacceptable to put this in the simulation model that which would represent the assembly line. More data got collected afterwards. This took a lot of time from the project because the new data needed to be added with the others that were already analyzed and calculated. It was not only one data sheet, it was three different data sheets that needed to be worked on again.

With analyzing and calculation means deleting all the bad times, adding times like average cycle time, average availability, average failures by calculation to get an input data for the simulation model and divide them into groups. By doing all this analyzation the excel files started to crash and it was impossible to make more than one move before the file crashed. There was a time when everything needed to start over from the zero because of the Excel errors. From that point, the data for different stations and failures got divided into several groups of excel files. This is where the macro programming started to do the work that facilitated everything.

Models

”Fast-Normal-Slow” was the only model that was built and got distribution data from expert fit. Problem with just one kind of model was limiting to do different kind of studies about the assembly line. As mentioned earlier, fast, normal and slow operators data got divided into 25% fast, 25% slow and 50% normal. The combination (of fast, normal and slow) model was created with distribution data from expert fit and also an empirical model. The data for the empirical model couldn’t fit the table file of plant simulation because it didn’t have more than two digits. The data for the empirical model got converted into percentages and all of the numbers were more than 4 digits and they needed to fit into the table file to get real results. So the problem got solved by doing over thousands of variants in excel file with data times to make it possible to put into the table file (plant simulation).

Created models were not exactly the same as the real assembly lines because of the functions of the plant simulation. Dismantle-station is a station that will divide in this case pallets which needed to be directly connected to assembly line stations but could not do that. This got solved by adding extra buffers in the model which make the created models not exactly similar to the real assembly line

So three reliable models were built that have the same characteristics as the assembly line in the factory with all the failures. By looking at the distribution and comparing them with each model, we all agreed that the “combination” model had the best with false chi statistics.

Something interesting that came up by analyzing fast, normal and slow operators that their distribution showed a true value. Table 7 is the distribution of a slow worker.

	string 0	real 1	real 2	boolean 3
string	Distribution	Chi statistic	Chi value	Result Chi
1	Gamma	9.8474	14.0641	true
2	Lognorm	9.9939	14.0641	true
3	Erlang	10.2804	14.0641	true
4	Logistic	19.5500	16.9175	false

Table 7. Data of slow workers.

Does it make the distribution more correct? Maybe not because this slow worker shown in figure 27 doesn't have as much data as "Fast-normal-slow" data or "Combination" data.

Future work:

The company needed a simulation model for the assembly line. In the future, these models could be used to test different methods and find new improvements in the line. With the data excel file made with macro programming will be easier for the engineers to put new data which will be customized automatically and it can just be added in the simulation model. Simulation models would also benefit the company from a sustainability perspective. By solutions to problems and finding improvements to increase efficiency and eliminating losses the economic will be better. This is also good for ecological perspective because, with the good production system, fewer materials will be wasted. We hope our project and the models we have created will help the company with development and growth.

Reflections:

The project was tough but very funny every new challenge got managed in one way or another But the most amusing part was to succeed with the project, learning new things and run the project, last but not least had the chance to meet and talk with amazing people like Tommy Sellgren, Richard and Niclas etc.

10. Conclusion

This section will explain the final and most important ideas that should be taken into account after having read the project completely. The reflection of the authors in the finalization of the project is clear, there is a great variability in the attitude of the workers and they affect to a large extent the production line, even though the line is working according to Lean techniques such as walking workers (WW), which allows the slower workers to distribute the bottleneck in a more homogeneous way throughout the line, reducing the importance that would have for the predecessor station if it is in a fixed position, to balance the line more easily and adjust the number of workers to satisfy the demand instead of needing a worker for each station. There is a takt-time that causes the group of six workers of the same line to have to wait for the operator slower, for that reason it is necessary to fight against these variations of attitudes. In the first place, it should be avoided that there are workers who produce at a production rate significantly slower than the rest of the team, for this the company must invest in intensive training, which should be carried out by the production leaders of each line teaching the new operators how to perform the work in the most efficient way, but if this is not possible, the variation of time should be avoided so that efficient workers are not hindered by the slowest operators. For this the proposal made from this project is that the **workers must be divided into working shifts** according to their condition, so the fast workers are reunited in the same shift, being able to work efficiently, and the slow workers are reunited in another shift so that although they work a little slower the variation of times will be lower. This is because the differences in time between the slowest workers is not as great as the difference in time between the slow workers and the fast ones.

Finally, it should be considered the automation of certain processes and guided tools that eliminate the difference of behavior among operators in this way, the variation of time will be less, there will be fewer errors of production so that the quality of the final product will be greater and the training times will also be lower since a guided machinery facilitates the completion of the work.