Design, analysis and optimization of a digital model of industrial production flow

Sergio Mora Carrión

LUND
UNIVERSITY

Department of Automatic Control

MSc Thesis TFRT-6144
ISSN 0280–5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Media-Tryck.
Lund 2021

Abstract

Digital twins are a very essential tool widely used by manufacturing companies nowadays. The scope of this project is to build a digital model of a real production flow present in SWEP's factory in Landskrona. The chosen software to accomplish the project is Tecnomatix Plant Simulation. Each module is constructed with real cycle times, OEE figures, velocities, and changeover times corresponding to the heat exchangers that are processed. Moreover, changes in the raw materials, tools, and orders are taken into consideration. To obtain a realistic behavior, real production orders are the input for the digital model. The simulation model can calculate different production parameters such as throughput times, waiting times, utilisations, or bottlenecks, among others. This digital model is also used to optimize the performance of the manufacturing process by changing the sequence of the processed jobs and it is utilized to explore potential improvements of its real twin. This is performed using advanced tools such as Genetic Algorithms or multi-level experiments. Finally, changes in the production flow are carried out to explore potential improvements in the mentioned facility's productivity, flexibility, and robustness. Some of the experiments show substantial productivity improvements and will be evaluated economically.

Sammanfattning

Digital twins är ett vanligt förekommande verktyg som används av företag idag. Projektets omfattning är att bygga en digital modell av ett verkligt produktionsflöde som finns i SWEPs fabrik i Landskrona. Den valda programvaran för att genomföra projektet är Tecnomatix Plant Simulation. Varje modul är konstruerad med verkliga cykeltider, OEE-värden, hastigheter och omställningstider kopplade till värmeväxlarna som bearbetas på fabriken. Dessutom tas det i hänsyn förändringar av råvaror, verktyg och order. För att få realistiska resultat är verkliga produktionsordrar använda som indata för den digitala modellen. Simuleringsmodellen kan beräkna olika produktionsparametrar, t.ex. genomströmningstider, väntetider, utnyttjad tid av maskinerna eller flaskhalsar med mera. Den digitala modellen används också för att optimera processens prestanda genom att ändra sekvensen för de bearbetade jobben och används för att utforska potentiella förbättringar av dess verkliga tvilling. Detta utförs med hjälp av avancerade verktyg som Genetic Algorithms eller simulationer med flera variabler som indata. Slutligen genomförs förändringarna i det verkliga produktionsflödet för att undersöka potentiella förbättringar i produktivitet, flexibilitet och robusthet. Några av experimenten visar betydande produktivitetsförbättringar och kommer att utvärderas ekonomiskt.

Acknowledgements

First I would like to express my gratitude to my supervisors at LTH and SWEP, Anders Robertsson and Lennie Liegnell. For their help and for giving me the opportunity to do this Master Thesis.

Secondly, to Magnus Wiman and Peter Nilsson. For their enthusiasm and for providing me with all the input I needed to conduct the project.

Thirdly, to Michael Herbig. For allowing me to work in his office, for the fikas we shared, and for all the interesting conversations we had about industry, economy and technology.

Finally, to my family and Magdalena. For being always by my side and for believing that some day I would achieve my dream of becoming an engineer.

Contents

List of Figures

Abbreviations

DT	Digital Twin
IoT	Internet of Things
TPS	Tecnomatix Plant Simulation
AI	Artificial Intelligence
BPHE	Brazed Plate Heat Exchanger
DES	Discrete Event Simulation
PLM	Product Lifecycle Management
GUI	Graphic User Interface
GAs	Genetic Algorithms
EA	Evolutionary Algorithm
HE	Heat Exchanger
PL	PresslLine
ISP	Integrated Start Plate
CHPL	Channel Plate
IEP	Integrated End Plate
MU	Mobile Unit
OEE	Overall equipment effectiveness
AGV	Automated Guided Vehicle

1

Introduction

The transformation of modern manufacturing facilities has been a very discussed topic lately. Market demands are very different and they require individual attention, customized products, high quality and short lead times. [8] These fluctuating demands have completely modified modern factories, transforming them into smart systems that are both faster and more flexible. Since modern facilities are increasingly getting more complex, companies need to optimize and test different configurations efficiently and safely to meet these new requirements. A very interesting way to accomplish this task is by constructing digital twins of the processes. The development of this tool provides the company with a safe and virtual space for testing and validation [11] .

1.1 Motivation

Simulation modeling solves real-world problems safely and efficiently. It provides an important method of analysis that is easily verified, communicated, and understood. Productivity urges to be optimized and testing innovative ideas may be risky, expensive, and, in the worst case, may require the system to stop. Digital twin (DT) is one of the most promising enabling technologies for realizing smart manufacturing and is characterized by the seamless integration between the cyber and physical spaces [2]. Competitivity among big producers is also boosting and to avoid becoming antiquated, DTs development is highly recommended. DTs are at the forefront of the Industry 4.0 revolution facilitated through advanced data analytics and the Internet of Things (IoT) connectivity and allow testing new adds-on or configurations without taking risks in a very affordable way. This appliance can be also interesting to test the production plans beforehand and obtain data from the manufacturing chain. With this, companies will be able to run cost analyses in the production process and foresee the cost of producing specific batches that require extra duties. Some instances are orders that call for specific painting, materials or weldings. The applications of the DT concept help reduce resource downtime, improve product throughput and quality, reduce manufacturing costs, and ensure operation safety [4]

1.2 Objective

This project aims to build and validate a DT of a real production flow present in SWEP's factory in Landskrona. After completing these tasks, the model will be used to obtain production throughputs such as capacities, utilization, or buffer sizes. It will be also utilized to improve the behavior by changing the sequence of the processed jobs or other production parameters. The chosen software to complete the project is Tecnomatix Plant Simulation (TPS), and this will be described thoroughly in section 2.5. This powerful tool offers a broad variety of functionalities that allow the user to create very sophisticated models. A featured utility is the Genetic Algorithm Wizard that will be used to re-schedule the production orders making the system faster. All the experiments conducted will be detailed in Chapter 4.

1.3 Methods

As an introduction to the problem, a review of the existing literature was conducted. This included relevant topics such as Industry 4.0., the benefits of using simulation and the state of art of digital twins. This first step motivated the choice of this project and proved its potential. TPS was the chosen software to accomplish the project, for this reason, an appraisal of its basic contents was carried out to justify its choice. A deep understanding of the production flow and the manufactured products was necessary to start with the simulation process. Once the simulation model was completed, it was verified and compared with the real plant to prove its capability. A set of different tests were performed during this verification process. A detailed set of experiments was designed to enhance the behavior of the model and were, at last, tested and evaluated with real figures.

1.4 Limitations

As in any simulation project, limitations exist and need to be considered. In the words of George E. P. Box, "all models are wrong, but some are useful". Throughout this project, limitations with different natures arose. Those limitations can be classified into three fundamental groups: the complexity of the model, unforeseen phenomena and time constraints. The first group includes some simplifications that were assumed in the simulated model to make the construction process a bit lighter. Some examples are:

- Use of waiting times to compensate for some actions of the robots that were not possible to simulate.
- The utilization of transfer stations instead of robots in some parts of the model, making the logic easier.

- Adaptation of the unbatching process.
- Consideration of the testing stations as black boxes from the simulation point of view.

The second group includes unforeseen events or changes in the production plan. An example is the omission of rebrazing processes in the line, with the latter being done with those units that do not clear the tests. Moreover, the production plan is considered fixed, whereas in reality is variable and can be adapted to execute urgent orders. Finally, some time constraints also affected the scope of the project. Only the continuous flow of the plant was considered, excluding some products not commonly produced and elements of the system. These elements added special complexity and variability to the process without providing extra added value. It is important to link this section with the last one of the report. Future considerations to upgrade the capabilities of the model are described, and it will be mentioned which ones would be more relevant.

2

Background

In the Fourth Industrial Revolution, the digital and the physical worlds are converging into a common point. They both need each other to continue developing and it is impossible to think of a factory without control systems, databases, or robots. Terms such as the IoT, Artificial Intelligence (AI), or Smart Factories have become part of our daily used vocabulary, but what do they stand for? An introduction to the company and a brief overview of the most relevant concepts and a theoretical framework is carried out in this section.

2.1 SWEP

SWEP is a world-leading supplier of Brazed Plate Heat Exchangers (BPHE) for heating, ventilation, air conditioning and industrial applications. Designed to make the most efficient use of energy, material, and space, the technology is quickly winning ground around the world, with SWEP at the forefront of developments. The company, based in Landskrona, was established in 1983 by two Swedish entrepreneurs and nowadays counts with more than 1000 employees. Moreover, it has production sites in five countries and representation in more than 50. SWEP offers the widest BPHE range on the market, from the smallest sizes to the highest capacities, matching the needs of both sensitive and aggressive processes, and with third-party approvals for performance to trust [13].

Brazed Plate Heat Exchangers

BPHEs are the smarter way to transfer heat. Their robust construction, with no gaskets or loose parts, improves operational reliability. The compact size means easy handling and flexible system design. Efficiency is high, with 95% of the material used to transfer heat, low hold-up volumes, and the ability to exploit small temperature differences. Those devices are basically composed by steel plates with pressed patterns that allow the exchange of heat between more than one fluid without mixture. These plates are brazed in furnaces to avoid leakages and provided the system with mechanical resistance. The behaviour of this device can be seen in figure 2.1. [13]

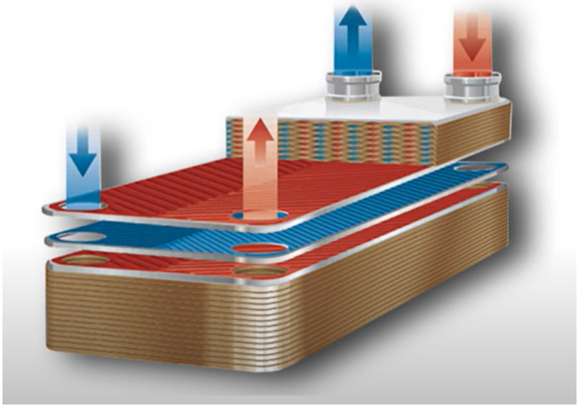


Figure 2.1: Brazed Plate Heat Exchanger

2.2 Industry 4.0

In recent years, Industry 4.0 is one of the most discussed subjects in production engineering. It started in Germany in 2011, and its development was possible due to the adoption of innovative systems in the manufacturing processes. Those were extremely expensive in the past, but now they became affordable and most of the leaders of the sector already have them present in their production plants. This transformation is based on the use of Cyber-Physical Systems and Information and Communication Technologies manufacturing systems, particularly, AI and the IoT. [3] This transformation strives for the creation of more flexible manufacturing systems that can adapt to the changing trends in the market by themselves by taking autonomous decisions.

2.3 Simulation

Simulation modeling is an excellent tool for analyzing and optimizing processes: when mathematical optimization of complex systems becomes infeasible, and when conducting experiments within real systems is too expensive, time-consuming, or dangerous, simulation becomes even more powerful. The simulation aims to support objective decision-making through dynamic analysis, to enable managers to safely plan their operations, and to save costs. [9] In general, to simulate means to mimic reality in a certain way. Simulations can be done, for example, through physical models such as wind tunnels, through simulators, or by using computer-based models for the evaluation of a given technical system or process design. In the last case, specific software is used to create a computer model that imitates the real-world process. The rapid development of computer hardware and software in recent years made computer simulation an effective tool for process modeling and an attractive technique for predicting the performance of alternative designs. It can also help optimize their efficiency. [6] Simulation can be defined in many ways; two interesting definitions are the ones given by Robinson (2015) and Law (2014):

"Experimentation with a simplified imitation (on a computer) of an operations system as it progresses through time, for the purpose of better understanding and/or improving that system."[10]

"In a simulation, we use a computer to evaluate a model numerically, and data are gathered in order to estimate the desired true characteristics of the model."[7]

Creating a simulation model is an iterative process. First, some sketches and a rough draft of the process must be developed. With this first approach, some results are obtained and can be compared to the real process to see the overall performance of the simulation. From this point, different iterations are carried out until the model copies the system as faithfully as possible. Building a simulation is quite time-consuming and at times may not be necessary, however, most of the time and efforts put in this kind of project go into understanding the process and its operation. Despite its many virtues, simulation is sometimes met with skepticism. Limitations exist and need to be considered. Nevertheless, in most cases, even a simplified model can be really helpful for the organization.

2.4 Digital Twins

A DT is a digital entity supported mainly by data. It can be described as a digital representation of the physical object or set of objects that are considered. This technology is divided in three parts, the real, the virtual, and the interaction between those two [5]. In this project, only the digital part will be considered.

The term was introduced in early 2000´s by Michael Grieves. In 2011, implementing DT was considered a complex procedure, which required many developments in different technologies [14]. However, with the development of technologies like cloud computing, Iot or the Industry 4.0 itself, it has become a more accessible task [1]. Nowadays there are several software widely used to create DTs, the chosen one to develop this project is described in the following section.

2.5 Tecnomatix Plant Simulation

Overview

Tecnomatix Plant Simulation is an object-oriented 3D software used to simulate, model, and animate complex systems and then test their properties and optimize their performance. It uses discrete event simulation (DES). A DES program only takes into consideration some points in time known as events. Those events are of importance to the further course of the simulation and the rest of the intermediate movements in between two of these events are not considered. Some instances of events are a part leaving a station, a robot loading a part onto a transporter, a part touching a sensor, etc. It may be possible to think that with this strategy, a lot of information will be lost. However, it is interesting to use it in this project due to the fact that by using DES, it is possible to simulate long operating times in just some minutes. This is especially useful when small changes in the configuration need to be tested or when it is necessary to replicate experiments. [12] This application is provided by Siemens Product Lifecycle Management (PLM) Software, a German company that is the main supplier of software for PLM and Manufacturing Operations Management. The software provides a quite intuitive Graphic User Interface (GUI) and preliminary models can be created in a relatively easy way by dragging and dropping the desired components. Even though this software is quite interactive, at times it is hard to reach the desired behavior. Advanced programming skills and previous knowledge of the tool are required if sophisticated models want to be created. Since every production system is very different from each other, it is hard to standardize working stations or processes in the software. Nevertheless, those tasks or requirements that cannot be created by using the predefined compo-

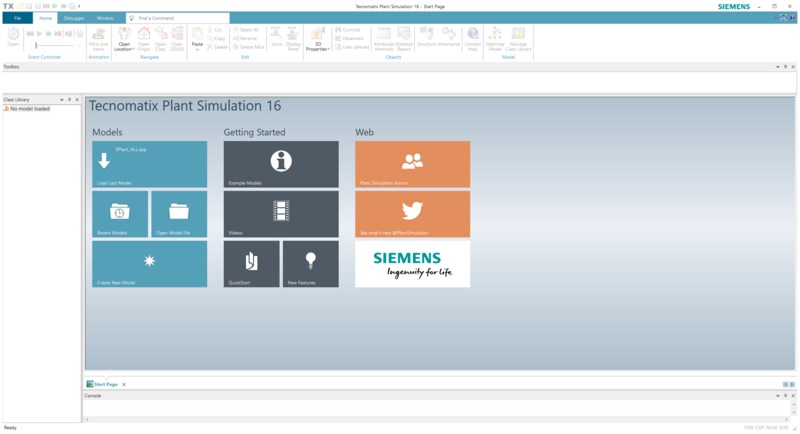


Figure 2.2: Tecnomatix Plant Simulation Start Page

nents can be, in almost all cases, be hardcoded. This fact makes the software very adaptable and useful in different contexts, but also a bit less accessible for beginner and inexperienced users. TPS counts with an array of analytic tools that can be used to extract data and draw conclusions from the simulations run. Some of those are the Bottleneck Analyzer, Statistical Charts, the Genetic Algorithm (GA) Optimizer or the Layout Optimizer. By using the Bottleneck Analyzer, it is possible to detect where those are located while running the simulation. This tool also creates a chart at the end of the run with the utilization of the different parts of the DT. The GA Optimizer, known as GA Wizard, makes it possible to optimize the simulation by using this mathematical tool. In this project, it will be used to re-schedule the production plan and perform the simulation in the lowest possible time. The Layout Optimizer enables minimizing transportation costs by using also GA. This tool is not of interest in this project since almost the whole layout of the plant is automated and the transportation of goods is done using conveyor belts. Finally, the Experiment Manager creates several scenarios and runs a set of simulations with different inputs such as processing times, availabilities or failures. At the end of the simulation, those results can be compared to detect if the changes made have affected the performance. TPS uses object-oriented programming. For this reason, some of the features of this programming paradigm must be mentioned. It is possible to create new classes from the existing ones and set them with the same properties. This is known as inheritance, and by creating a base class and deriving it, other instances can inherit the base characteristics. Those classes can be defined in different ways to create more complex systems with an easier structure. This feature is known as polymorphism. Finally, another interesting feature typical of object-oriented pro-

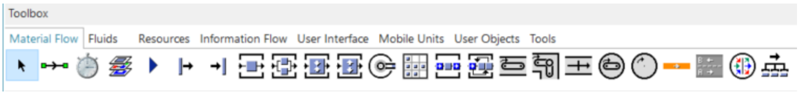
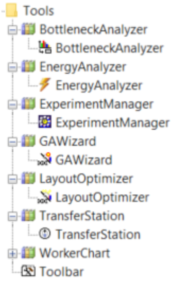


Figure 2.3: Tecnomatix Plant Simulation Toolbox



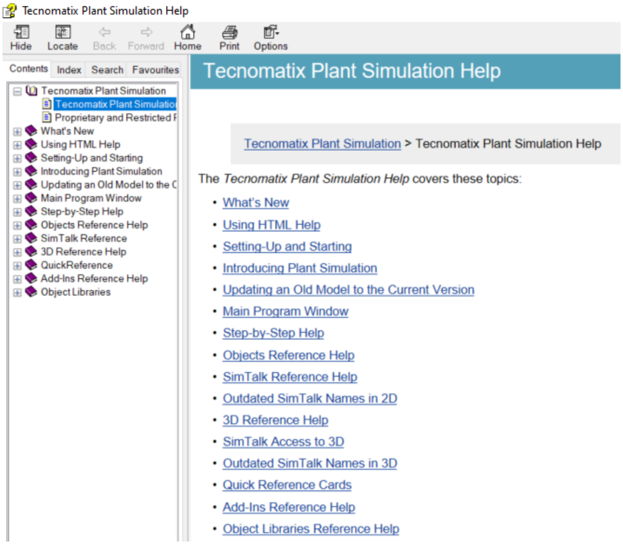
Tools
- BottleneckAnalyzer
  - BottleneckAnalyzer
- EnergyAnalyzer
  - EnergyAnalyzer
- ExperimentManager
  - ExperimentManager
- GAWizard
  - GAWizard
- LayoutOptimizer
  - LayoutOptimizer
- TransferStation
  - TransferStation
- WorkerChart
- Toolbar

Figure 2.4: Tecnomatix Plant Simulation Tools

gramming is the hierarchy. This allows us to create complex models and separate them into several connected levels, making them easier to understand and debug.

Practical Considerations

TPS counts with a very intuitive help tab. This tool contains all the required information to become quite skilled with the software and makes the learning process faster. All the methods and additional information are gathered there and it is possible to find the required information by merely typing the command in the index tab. As mentioned before, advanced programming skills are required if complex models are constructed, thing that makes the software both more flexible and a bit more difficult to use for beginners.



Tecnomatix Plant Simulation
Tecnomatix Plant Simulatio
Proprietary and Restricted F
What's New
Using HTML Help
Setting-Up and Starting
Introducing Plant Simulation
Updating an Old Model to the C
Main Program Window
Step-by-Step Help
Objects Reference Help
SimTalk Reference
3D Reference Help
QuickReference
Add-Ins Reference Help
Object Libraries

Tecnomatix Plant Simulation Help

Tecnomatix Plant Simulation > Tecnomatix Plant Simulation Help

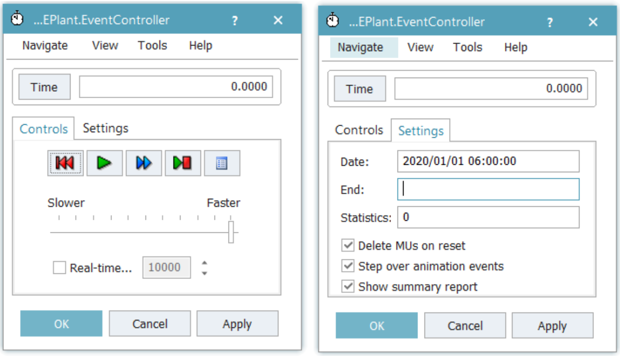
The *Tecnomatix Plant Simulation Help* covers these topics:

- What's New
- Using HTML Help
- Setting-Up and Starting
- Introducing Plant Simulation
- Updating an Old Model to the Current Version
- Main Program Window
- Step-by-Step Help
- Objects Reference Help
- SimTalk Reference Help
- Outdated SimTalk Names in 2D
- 3D Reference Help
- SimTalk Access to 3D
- Outdated SimTalk Names in 3D
- Quick Reference Cards
- Add-Ins Reference Help
- Object Libraries Reference Help

Figure 2.5: Tecnomatix Plant Simulation Help Tab

The programming language that is integrated into TPS is SimTalk. SimTalk is tightly integrated with the simulation objects and allows accessing methods and attributes of the built-in objects of the model. Like other programming languages, it also includes control structures such as loops and conditional branching. To achieve a more realistic representation, TPS can import models with jt format and animate some parts of those to include them in the simulation. In addition, it is possible to import, and also export data to other systems, databases, or programs such as Excel, Oracle, or AutoCAD.

To run a simulation, an event controller is required in the model. This can be set to finish the simulation after a certain period and can be stopped or started using the available buttons. A picture can be seen in Fig. 2.6.



Time 0.0000

□ Real-time... 10000

Date: 2020/01/01 06:00:00

End:

Statistics: 0

☑ Delete MUs on reset

☑ Step over animation events

☑ Show summary report

(a) Event Controller Controls (b) Event Controller Settings

Figure 2.6: TPC Event Controller

2.6 Genetic Algorithms

Multi-level experiments often lead to a huge number of simulation runs, especially when considering several combinations of a large number of inputs. By the application of the so-called Genetic Algorithm, the absolute number of experiments carried out is reduced considerably while there is still a chance of finding good solutions. [9] GA is a biology-based general optimization technique and a particular class of evolutionary computation algorithms. Its origin is based on the process of natural selection: individuals who adapted best to their environment, procreate the best. These algorithms are widely used by scientists and engineers in many different fields such as computer science or operations research, and belong to the larger class of Evolutionary Algorithms (EA). It is important to mention that GAs provide good solutions for complex problems but optimality is not guaranteed. The GA algorithm starts with an initial population of solutions (i.e., configurations of input parameters). An individual is the name used to designate a solution together with the corresponding solution value. The solution itself is called a chromosome. A fitness value is defined to denote the solution value. Another definition that must be introduced is the so-called generation. A generation is an iteration to create a new population. With the initial population, solutions are combined to obtain the next generation of, hopefully, better solutions. This is known as offspring while the previous generation is denoted as parent. The generation of this new solution is carried out by using selection, crossover, inversion, and mutation, just like nature does. [9]

Those consist of:

- Selection: the fitter an individual solution the more times it is likely to reproduce.

- Crossover: random exchange between two chromosomes to create offspring.

- Inversion: inverting a chromosomes' sequence.
- Mutation: random flipping within a chromosome.

The algorithm will stop when the maximum number of generations has been passed or when the tolerance threshold is reached. In this project, GAs are utilized to reschedule the production orders improving the execution time of the required orders. This will be explained in more detail in sections 4.3 and 5.2.

3

Construction of the Digital Model

This section contains a detailed description of the most important steps that were completed in the construction of the DT. At first, a review of the production line, the produced goods and the raw materials serves as an introduction. After this preface, the logic division of the total flow is explained, and the construction of those different systems is illustrated. The coding of the desired behavior and the configuration of the diverse elements are then described in detail. As mentioned in previous sections, real production orders are the input of the model, with the former being also elaborated in this chapter. Finally, the steps towards the fine-tuning and the validation of the model along with the data acquisition procedure are defined and explained.

3.1 The Production Line

The production line at issue is present at SWEP's factory in Landskrona, Sweden. The so called E-class production line is capable of producing a wide range of heat exchangers (HEs) with many different configurations. Those that are relevant for this project will be detailed below. As mentioned before, this project only involves the automated part of the line. Therefore, the start and end plates manufacturing will not be part of the DT as they are produced in another press line that is not part of the continuous flow. The production of specific products that require another press line (PL) or manual assembling will not be considered either. As mentioned in previous sections, the consideration of these added complexities was not in the scope of the project. The production starts by pressing and cutting the channel plates. Those consist of two overlapping layers of copper and steel and are loaded onto a small pallet at the end of the PL. This pallet has previously been loaded with the Integrated Start Plate (ISP), which is also made of copper and steel. Once the desired number of channel plates (CHPLs) has been loaded, the pallet is transferred to a conveyor, and then the Integrated End Plate (IEP) is loaded, forming the HE that only needs to be brazed. In the case of the IEP, steel is the only material used. As a special feature, the PL produces two CHPLs per stroke. For this reason, two pallets are loaded simultaneously, which increases the speed of the whole process. The HEs are then transported to the most complicated and sophisticated part of the assembly line. A cell with four robots prepares the batches and delivers them to the furnace where the HEs are brazed. Working in a very coordinated way, the robots place eight HEs and then eight intermediate fixtures alternatively onto a roster, forming a solid set ready to be brazed. After this process is completed, another robot deposits the pressure plates and brings the batch to a screw table where the required force is applied for the HE to be brazed correctly. The number of layers per batch depends on the quantity of CHPLs of the corresponding products, and this figure is established by the furnace. Loading too many products could result in the batches being too high and not fitting in the furnace. The usual number is between one and four. Once the batches are ready, a continuous furnace brazes the HE in a process that lasts two and a half hours. To maintain the concentration of gases and the temperature, the furnace is only opened every six minutes, which restricts the number of rosters that can be inside simultaneously. After the brazing is completed, the rosters come back to the above-mentioned cell through conveyor belts. This cell also unloads the rosters after the brazing process. Similarly, the robots first unscrew the pressure plates and then place the HE in the corresponding conveyor belt. These will bring each unit to a testing station, which guarantees the quality of all the products. A schematic drawing of the complete layout can be seen in Fig. 3.2.

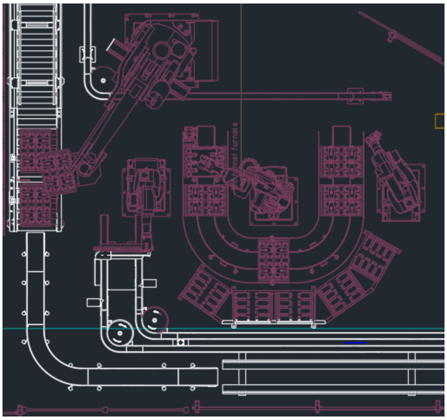


Figure 3.1: Robot Cell Schematic

Products Realized

One of the most notable characteristics of this line is the adaptability to different products, materials, and configurations. Even though there are several of them, the

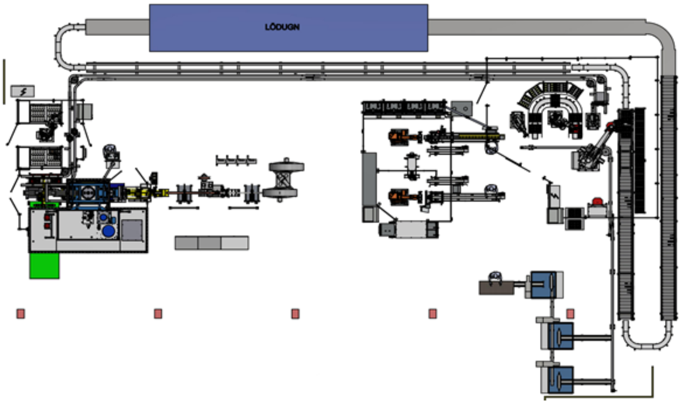
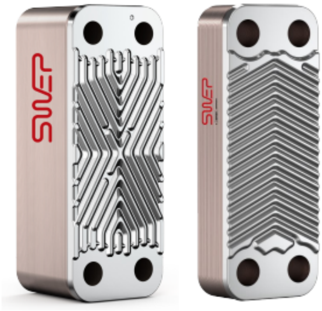


Figure 3.2: E Plant Schematic

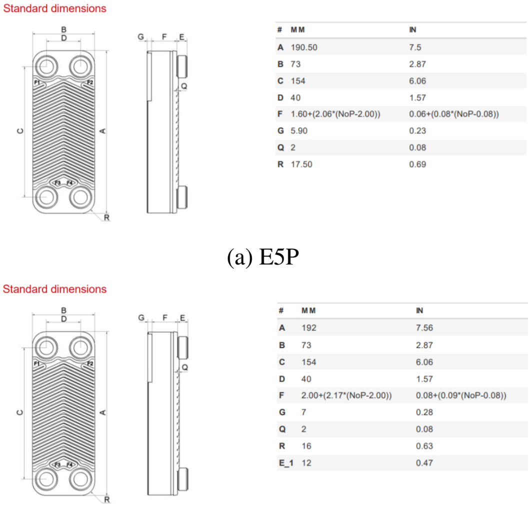
majority of the orders are different configurations of two models of HE, the E5P and the E5AS. Those can be seen in the figure 3.3 and can be built with different number of CHPLs, being the usual figure between eight and sixteen. Their dimensions and



(a) E5P (b) E5AS

Figure 3.3: Most Produced Heat Exchangers

shapes can be seen below. The length of both is under twenty centimeters, and they belong to the smallest SWEP's product range. With regards to the materials, a wide variety of combinations exist. As mentioned before, the HEs are produced with only steel and copper, but those have different compositions and thicknesses. In the case of the continuous flow, a total of seven different coils require to be considered. Four of those will be steel coils and the three remaining copper. By combining them, a total of six different HEs are manufactured. All of those will use different pressing and cutting tools in the PL. These changes are also considered in the DT, adding variability and hence, complexity.



#	MM	IN
A	190.50	7.5
B	73	2.87
C	154	6.06
D	40	1.57
F	1.60+(2.06*(NoP-2.00))	0.06+(0.08*(NoP-0.08))
G	5.90	0.23
Q	2	0.08
R	17.50	0.69

(a) E5P

#	MM	IN
A	192	7.56
B	73	2.87
C	154	6.06
D	40	1.57
F	2.00+(2.17*(NoP-2.00))	0.08+(0.09*(NoP-0.08))
G	7	0.28
Q	2	0.08
R	16	0.63
E_1	12	0.47

(b) E5AS

Figure 3.4: HEs' Dimensions

The Division of the Production Line

Understanding the facility to model is fundamental for creating a useful simulation model. It is not possible to reach a similar behavior in the DT if the whole system with its different parts is not deeply comprehended. In this section, the logical division of the analyzed flow is presented and justified.

Breaking down systems or problems into smaller and simpler ones is a common procedure for engineers, scientists, or other professionals. Following this philosophy, it was decided to separate the line at issue into various subsystems. This division was used when creating the DT to validate and check that the elements acted as desired. The parts into which the plant was divided are:

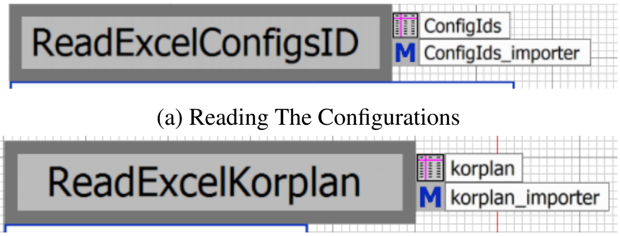
- Press Line.
- Loading Cell.
- Robot Cell, Furnace, and Conveyor Structure.
- Tester Station 3.
- Tester Station 4.

3.2 Coding the Behavior of the Plant

This section aims to describe the construction of the DT. At first, an overview of how data is imported from Excel to the model is shown. Then, the development of the above-mentioned submodels is exposed and finally, the most important methods programmed are reviewed. A method in TPS is simply a script where the user can code the nature of the components of the DT. Methods can be used, for instance, to calculate parameters, to stop parts or to change the routing of the Mobile Units (MUs).

Importing Data From Excel

Since the beginning of the project, the aim was to create a standardized model simplistic enough to manipulate for both experts and new users. To make the DT flexible and adaptable to different settings and production plans, it was decided to import some information from external files. By only modifying some parameters in the Excel files, different behaviors of the plant can be simulated. In this model, two tables are imported to the model and are essential to run the simulation. The first one contains the main features of the PL, which is the most important element of the plant. Information such as cycle times, materials consumption, or identification numbers of the raw materials is included in this sheet. The other table contains the production plan with the different orders and other required information that will be detailed in section 3.3. The data is imported by pressing the buttons shown in figure 3.5 and selecting the desired Excel file. The methods are coded to import and paste in the tables the correct information with the right format so that the simulation can be run.



(a) Reading The Configurations

(b) Reading The Production Plan

Figure 3.5: Buttons to Import Excel Files

Press Line Configuration

The PL is the most important part of the model. The whole production is programmed for this machine and the rest of the downstream systems work as a consequence of its output. The PL subsystem is composed of several sources and buffers for the coils, working stations, and a buffer that will create (by coding) the corresponding parts and place them on the conveyor belt.

As mentioned above, the PL reads from the tables the main features that affect its behavior; ID of the tools and coils, products generated per stroke, or cycle times amongst others. Then, a generator triggers the method that produces the CHPLs. A generator is a TPS component that triggers the desired method every certain time interval. In this case, the triggered method will produce the CHPLs and subtract the corresponding weight from the coils, imitating the real consumption of materials. It will also check if those materials are over and trigger the corresponding changes, stopping the PL when this occurs. This method will be activated every certain interval. In this case, the takt time of the CHPLs. A counter for the tool's lifespan will be also incremented and will replace them when they require maintenance.

The project´s aim is to standardize all the created components to use them in future models. Therefore, most of the features can be modified through Excel files. Replacements of the coils and tools are considered, and to achieve this, a smart system had to be designed. At first, one coil of each kind is released and placed in a buffer. Then, the model reads the required material and opens the gate of the corresponding buffer. From there, the worker picks it up and places it in the station, where it will be processed. Once the coil is over, it is deleted and the same buffer is opened again to be able to place a new one. In case the coil is not over and requires to be changed, for example, when there is a change in the orders, it is brought back to the first buffer so that the remaining material can be used in future jobs. This system imitates reality quite faithfully, and the times to load the different materials correspond to the real ones. The different methods can be seen in the figure 3.6. The last feature included in the PL model is the test button and the calibration checkbox. Those tools were used in the verification process, and are useful to stop the PL when it is necessary or to run experiments independently of other subsystems. A whole picture of the PL model is shown in figure 3.7. This presents all the components that form the system without its real jt model on it.

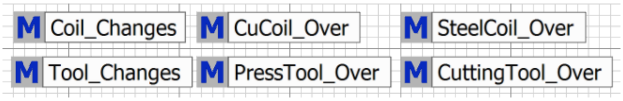


Figure 3.6: Mehods to Change Coils and Tools

Loading Cell Configuration

Once the CHPLs exit the conveyor of the PL, they are loaded on small pallets along with the ISP and IEP. At first, a transfer station places the ISP onto the small pallet coming from the other part of the production. Then, the CHPLs are pilled, and finally, the IEP is loaded, forming the HE ready to be brazed. The loading times are set in the transfer stations and correspond to the real ones. All the time mea-

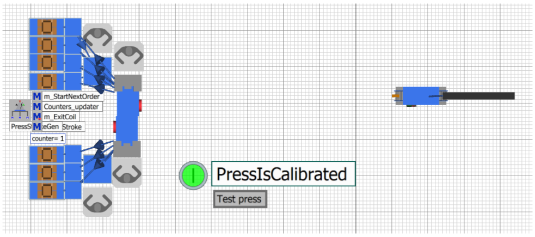


Figure 3.7: Press Line Model

surements in the real plant were taken scrupulously as an average of 5 different evaluations. In the case of the CHPL's loading process, this time is the same as the takt time of the PL. This is set in reality to coordinate the strokes and the loading process in order to avoid buffers. Since these times vary depending on the stroked product, they are also modified in the stations by using code. A picture of the whole subsystem is shown in Fig. 3.8.

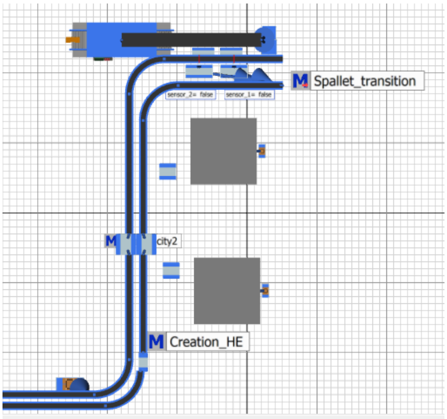


Figure 3.8: Loading Cell Model

Yet this operation is more complicated. Two products per stroke are produced, and consequently two pallets must be loaded at the same time. This behavior requires some coding and additional considerations. To achieve it, two sensors that are in charge of stopping the two pallets are placed on the conveyor. When CHPLs are available and these two pallets are ready, they both are moved to the station. Then, the stacking process begins. Different configurations exist, and the number of CHPLs per unit is set in this cell. By default, mobile units such as pallets are fully loaded before being released from the station. Accordingly, it was necessary to modify the capacity of the small pallets by coding. In this way, the exact amount is loaded. Then, just before entering the station where the IEP is placed, one of the

CHPLs is deleted to let the other part enter. Even though it may be thought that we are disconsidering the actual behavior with this strategy, it is a smart way of getting the desired performance. Just after being completely loaded, the HEs enter the last part of the cell. A working station governed by a method destroys all the plates and creates an instance of a HE. It also writes different attributes to guarantee traceability in the whole chain. This work has a duration of zero seconds, since it is just a fictional operation.

The Robot Cell Configuration

The units move through conveyor belts until they reach the robot cell, where batches of HEs are prepared and sent to the furnace. The configuration of this cell is quite complex and requires the use of robots, sensors, counters, conveyor belts and methods to program the entrances or exits of the units. As in the previous cell, real times were taken and added to the robots and conveyors respectively. The simulation model can be seen in the picture 3.9.

As mentioned, the HEs will form layers of eight. Hence, the production plan is always divisible by this number. However, the last of the rosters may have a lower number of layers than the others. This behavior is considered in this cell and it is programmed using counters and the above-mention attributes that are written in previous activities. Other counters will keep track of the batching process to move an intermediate or top fixture conveniently when a layer is completed. The rosters where the batches are formed are placed on the circular conveyor by another robot. This robot first detects if there is a spot available, and then proceeds to complete this movement. The same robot is also programmed to send the finished batches to the furnace. This logic is programmed using connectors between the different conveyors, and by employing different methods that govern the entrances and exits of the conveyors and the robot itself. A sensor checks whether the rosters are empty or full to stop them or let them pass. This feature is needed only for the first iterations when no products have yet been brazed and thus, empty rosters are moved.

The logic of the cell can be divided into two procedures, preparing the batches or disassembling those. Those two activities are narrowly related to each other: as soon as a roster is unloaded, it is used in the next iteration to prepare a new batch, thus acting in a cyclical form.

Batching: When an empty roster gets to the loading position, the first robot places eight units by using a counter. Then, this conveyor is blocked and the gate for the intermediate fixtures becomes open. A different counter is incremented every time a layer is completed, and when all of them are stacked, a top fixture is released. Then, the robot places the whole batch on a conveyor that leads to the furnace and increments a counter for these.

Unbatching: The unbatching responsibility relapses into the last robot. This robot grabs an item from the roster and, depending on its nature, places it on the conveyors that lead to the testers or back in the buffers where the fixtures are stored. This

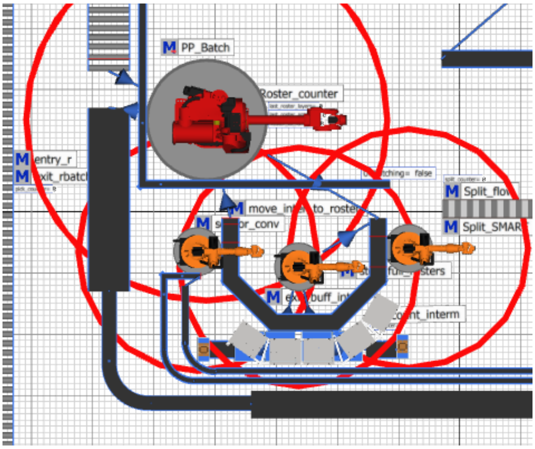


Figure 3.9: Robot Cell Model

desired action is programmed using a method. The tester units are open or close depending on the amount of CHPLs of the HEs at issue. When working with high amounts of those, fewer units are produced, and a lower testing capacity is required. These capacities are known and the testing units utilizations are:

- Less than sixteen CHPLs per unit, both testers.
- Between sixteen and twenty CHPLs per unit, only tester 4.
- More than twenty CHPLs per unit, only tester 3.

The decision taken by the robot that implies this logic is programmed using a method. In section 4.3, a new methodology to split the flow will be proposed and analyzed.

The Furnace Configuration

The furnace has been simulated as a conveyor belt with a specific feature, which is that its entrance can be opened only every six minutes. To achieve this behavior, a method to block the entrance was programmed. This method opens the gate and closes it for six minutes, making the other products wait outside until this time period passes. The velocity is set for the whole process to last two and a half hours.

The Testers Configuration

In this DT, the testing stations have been considered as black boxes for the simulation. Real takt times and capacities are set in these stations for the setup of the model. As a special feature, testers can be turned on and off with the buttons. This

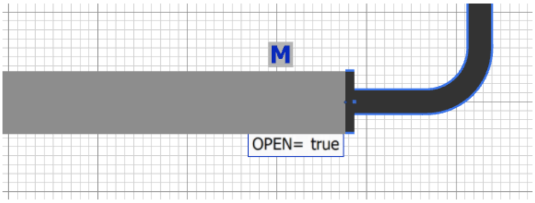
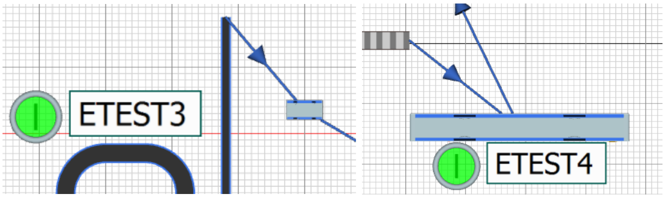


Figure 3.10: Furnace Entrance Model



(a) E Test 3

(b) E Test 4

Figure 3.11: Testing Station Models

is particularly useful when tests need to be run or when other products which cannot be processed in the continuous flow are being tested in these stations.

Once the products are tested, they are separated into two flows, one for the valid HEs and another one for the scrapped units. This second flow will be one percent of the total number of produced units.

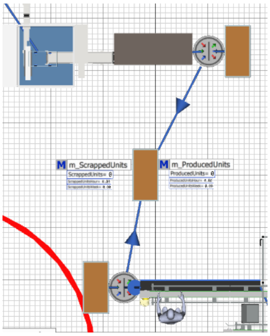
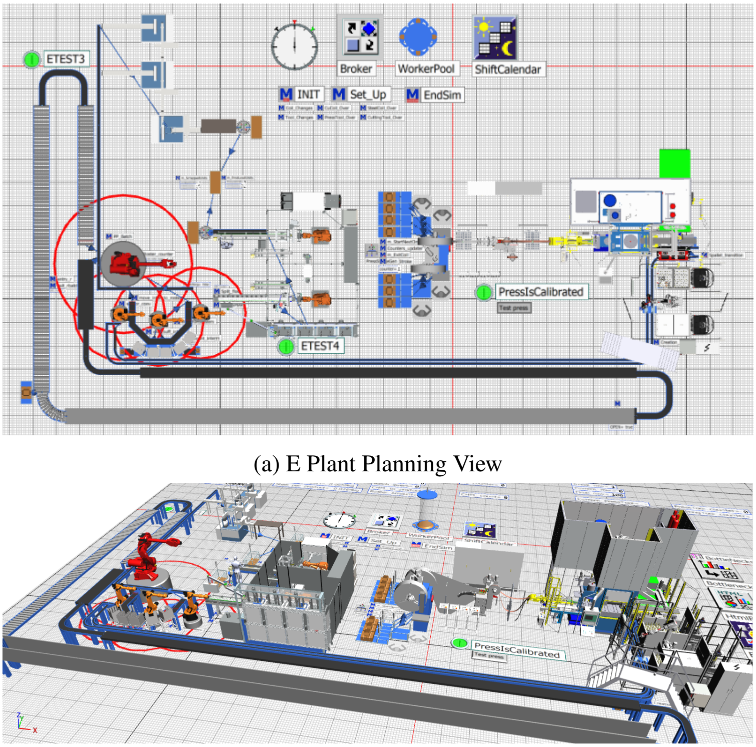


Figure 3.12: Scrapped Units Splitter

E Plant Model

After having discussed all the different subsystems that form the plant, it is necessary to expose briefly how all of them were merged to work as one. In this section, the integration of all the components, their communications and other overall capabilities of the plant are detailed. The whole layout can be seen below (Fig. 3.13).

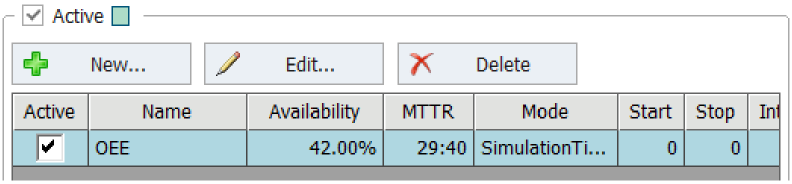


(a) E Plant Planning View

(b) E Plant 3D View

Figure 3.13: E Plant Model

With TPS, failures in different machines can be set to achieve a performance closer to reality. In this case, Overall Equipment Effectiveness (OEE) figures were set in the PL and the tester stations. These numbers were obtained from the real plant and were activated or deactivated depending on the experiment's aim. A mean time to repair for these failures was also set and calculated as an average of the stop time of the real system. OEE figures were adapted to the model. In reality, changes of coils and tools are included in this figure. However, it had to be extracted since the model considers those operations. The DT is built with real dimensions. All the conveyors, the furnace, and other components have the same dimensions that the factory has. In addition, speeds of the different elements of the production flow were measured and set in the setting tabs of the simulation elements.



Active	Name	Availability	MTTR	Mode	Start	Stop	Inf
☑	OEE	42.00%	29:40	SimulationTi...	0	0	

Figure 3.14: OEE PL Model

Flexibility was required for the model. For this reason, a capability to run a specific order or an array of orders was programmed. By changing two variables, the methods can read which line or lines of the production plant wants to be simulated.

Achieving full integration of the submodels was a challenge when building the model. They needed to communicate with each other in order to consider features such as batching layers, processing times, or materials. This problem was solved by adding features to all the components in the corresponding methods.

When a simulation is running, the console shows real-time information of the batch that is being produced. Again, this behavior was achieved through coding.

In the coming lines, specific features and methods of the E Plant DT are thoroughly explained.

Mobile Units: The MUs of the plant are elements that are used to simulate the flow of materials. They give a visual representation and make the model dynamic and easier to understand. In this particular DT, the MUs are created by sources or by the model itself (case of the HEs). They are listed below, and a picture of all of them can be seen in figure 3.15.

- Steel coils, four different identification numbers.
- Copper coils, three different identification numbers.
- Integrated End Plates.
- Integrated Start Plates.
- Channel Plates.
- Small pallets, forty units.
- Heat Exchangers, a total of six different models.
- Rosters, thirty-two units.
- Intermediate Fixtures.
- Top Fixtures.

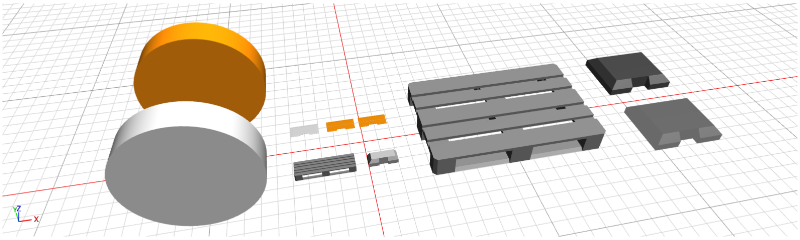


Figure 3.15: Mobile Units

The Init Method: The Init Method is run every time the simulation is started. It contains the fundamental setups of the plant and reads the information from the tables to start a new order. At the end of the code, it triggers the method “Start Next Order” and prints a message to indicate that the initialization was done correctly.

The Start Next Order Method: The Start Next Order Method is one of the most important methods of the DT. Even though it belongs to the PL subsystem, it is fundamental to reach the whole performance, and for this reason, it will be explained in detail in this section. In the first iteration, the method is triggered by “INIT”, and in the following ones by the stroke generator itself. This method resets all the counters so that the next order can be started. The next step is choosing which line of the production plan is going to be executed and obtaining the configurations for this type of HE. Once this task is finished, it sets the processing times for the Stroke Generator and other stations of interest and prints the basic information of the job to produce. To load the small pallets correctly, their dimension is set here, and then the corresponding tool and coil changes are executed. If the first iteration is executed, a setup of the different elements is completed. After checking that the materials were loaded correctly, the PL is calibrated and thereby initialized. The last task of this method is the calculation of the amount of HE that will be batched in the last roster. As explained above, it will be a multiple of eight, but it can contain a lower number of layers. A final message is written in the console to let the user know that the order was initialized correctly.

The EndSim Method: The EndSim Method is run at the end of the simulation, either when the maximum time is reached or when the simulation is stopped by using code. The latter may happen if the whole production plan has been completed. This method calculates interesting parameters of the simulation that will be collected in an HTML Report. Those outputs are materials consumptions, time spent in changing coils and tools, and units and CHPL per hour and week. Additionally, it filters the result table from the Bottleneck Analyzer to show only the most interesting parts of the system. When running the Bottleneck Analyzer, all the equipment is tested. However, elements such as conveyors are working all the time even if they work as buffers or if they are empty, making the result not realistic. Finally, the EndSim Method triggers the generation of an informative report. This will be analyzed in the next section.

Broker, Worker Pool, and Shift Calendar: These three blocks are necessary when working with operators.

The Broker is the go-between for services offered and services required. It is needed to add services such as the mounting of the coils.

The Worker Pool generates the worker for the simulation. In this case, since almost the whole production is automated, only one worker is generated.

Shift Calendars are used to simulate real shifts in factories. In the case of the E Plant, the flow is continuous and works 24 hours a day during the seven days of the week.

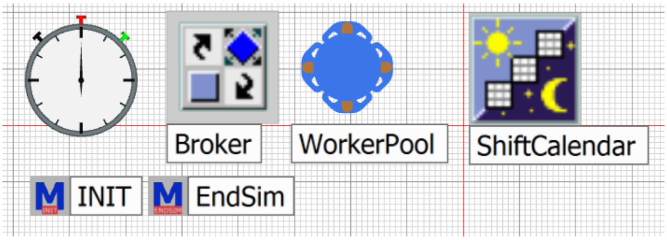


Figure 3.16: Methods and Worker Resources

Data Acquisition: After having seen the construction of the model, it is important to gather the desired outputs and visually present them. This process is extremely important, because if data is not gathered and analyzed, it is not possible to evaluate the changes done in the flow or the correct behavior of the DT itself. TPS allows the user to extract and display data from the simulation in different ways. In this project, several tools are used. The first one is the use of data charts. Those can be seen in figure 3.17 and configured by dragging and dropping the desired parts onto them. In the case of the furnace, the chosen feature was occupancy instead of the default settings. These show waiting, blocking, and working times mainly.

Another utility show is the HTML report block. By coding this script using HTML language, it is possible to generate the desired output. The report in this project shows an overall introduction, a picture of the model, the statistic blocks, production data (capacities per hour and week), consumption of materials, throughputs, changing times, and the results from the Bottleneck Analyzer. An example can be seen in the appendix A.

The Bottleneck Analyzer generates a table with the utilization of all the components of the DT. By filtering these results, conclusions can be drawn to improve some systems in particular. The GA Wizard integrates GAs into an existing simulation model. This tool is used and set up to optimize the production plan by changing the order of the jobs to produce. More specific information is available in section 4.

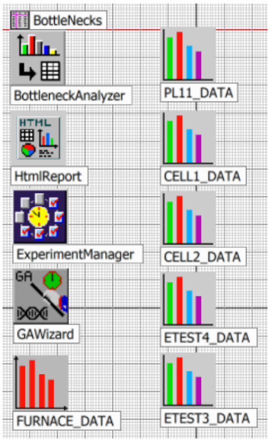


Figure 3.17: Data Acquisition

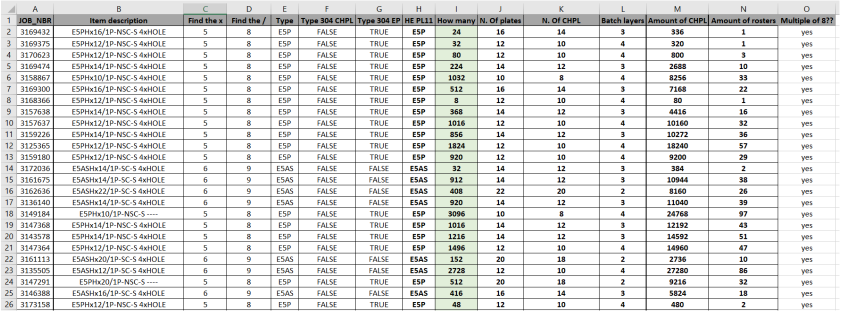
The experiment manager is a very convenient appliance to run experiments with different inputs. By setting the outputs and the inputs of the experiment, it is possible to measure the difference in the performance. A more complete report can be also generated (See Appendix D). Those experiments usually take a considerable amount of time. More information about its use in this project will be described in the section 4).

The topics introduced in this section will be detailed in section 4. There, the settings of the experiments are discussed.

3.3 Construction of the Production Plan

One of the most interesting capabilities of the model is the fact that it can import information from other software. In this case, Excel was the chosen programme to gather the production orders. Its versatility, compatibility with TPS, and its ease of use made Excel a suitable choice. For this reason, some calculations will be carried out in Excel sheets before being imported by the DT. In this section, the conditioning of the required tables and data is explained in detail. This step is vital due to the fact that if the information is not imported correctly the model will not be able to run. At first, the order ID is split to read exactly which product is produced and if the required material for the CHPLs is steel 304. A new code with the corresponding name is generated, and this is the cell that the DT will have as input. With regards to the quantity of HE to produce, this is calculated using the function ceiling. First, one percent is added to all the initial orders to consider the scrapped units. Then, the ceiling function is used to make this number divisible by eight. Another input that the model reads is the amount of CHPLs per job. This is essential to stop the PL when the order is completed. With the number of CHPLs per HE, it is also possible to calculate the layers that the batches will have. This is done by looking up the values in a table using the functions Index and Match. Finally, the total amount of required rosters for the order is calculated. With this, the model will be capable of adjusting the layers of products that the last roster will contain.

A picture of some lines of the conditioned plan can be seen in figure 3.18.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	JOB_NBR	Item description	Find the x	Find the /	Type	Type 304 CHPL	Type 304 EP	HE PL11	How many	N. Of plates	N. Of CHPL	Batch layers	Amount of CHPL	Amount of rosters	Multiple of 8??
2	3169432	E5PHx16/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	24	16	14	3	336	1	yes
3	3169375	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	32	12	10	4	320	1	yes
4	3170623	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	80	12	10	4	800	3	yes
5	3169474	E5PHx14/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	224	14	12	3	2688	10	yes
6	3158867	E5PHx10/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	1032	10	8	4	8256	33	yes
7	3169300	E5PHx16/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	512	16	14	3	7168	22	yes
8	3168366	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	8	12	10	4	80	1	yes
9	3157638	E5PHx14/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	368	14	12	3	4416	16	yes
10	3157637	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	1016	12	10	4	10160	32	yes
11	3159226	E5PHx14/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	856	14	12	3	10272	36	yes
12	3125365	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	1824	12	10	4	18240	57	yes
13	3159180	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	920	12	10	4	9200	29	yes
14	3172036	E5ASHx14/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	32	14	12	3	384	2	yes
15	3161675	E5ASHx14/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	912	14	12	3	10944	38	yes
16	3162636	E5ASHx22/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	408	22	20	2	8160	26	yes
17	3136140	E5ASHx14/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	920	14	12	3	11040	39	yes
18	3149184	E5PHx10/1P-NSC-S ----	5	8	E5P	FALSE	TRUE	E5P	3096	10	8	4	24768	97	yes
19	3147368	E5PHx14/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	1016	14	12	3	12192	43	yes
20	3143578	E5PHx14/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	1216	14	12	3	14592	51	yes
21	3147364	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	1496	12	10	4	14960	47	yes
22	3161113	E5ASHx20/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	152	20	18	2	2736	10	yes
23	3135505	E5ASHx12/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	2728	12	10	4	27280	86	yes
24	3147291	E5PHx20/1P-NSC-S ----	5	8	E5P	FALSE	TRUE	E5P	512	20	18	2	9216	32	yes
25	3146388	E5ASHx16/1P-SC-S 4xHOLE	6	9	E5AS	FALSE	FALSE	E5AS	416	16	14	3	5824	18	yes
26	3173158	E5PHx12/1P-NSC-S 4xHOLE	5	8	E5P	FALSE	TRUE	E5P	48	12	10	4	480	2	yes

Figure 3.18: Production Plan Excel

3.4 Validation of the Digital Model

Validation is a crucial step after constructing the whole DT. This must be performed carefully and scrupulously to assure the correct behavior of the model. If this part of the project is not done correctly, the whole behavior can turn out to be unknown, unrealistic, and thus not useful in real problems. Limitations and simplifications exist, and they are taken into consideration. Nevertheless, the model must imitate reality and similar results need to be outputted. In this last section of the construction of the DT, the validation process is detailed. Since all the subsystems have been constructed separately, and their validation processes were conducted in separated parts as well. However, after merging the whole system, a global validation and debugging stage were also required.

Press Line: The Press Line verification is performed by adding a drain at the end of the conveyor belt, by analyzing the total output and by adding counters for the CHPLs. The output verification is confirmed by comparing the figures from the simulation to the ones obtained theoretically and by using the different takt times. Replacements in the coils and the tools are considered, and therefore it is essential to check the correct performance. To execute this step, two production plans are prepared. The first one would contain a long batch of the same product so that deterioration in tools and materials' consumption could be confirmed. The second one would contain a wide variety of products and small quantities to produce. With this, many changes occur in the same simulation. These two tests confirm the correct behavior of the most complex part of the PL. The OEE for the PL is set to 42%. This figure is obtained from the real plant and after several calculations. In reality, the OEE figures include changeover times. Since the model can deal with those changes, those are extracted from the calculation of the total OEE.

Loading Cell: Real loading times and speeds are set in this cell. The verification step is simpler in that case, but it needs CHPLs and hence the PL. The verification of this subsystem is completed along with the PL. Again, real times are considered in the loading stations. Moreover, to keep the PL flow, those stations need to adapt their processing times to the takt time of the previous system. As a consequence, code is used to change this feature at the beginning of every order. The placement of ISP and IEP is validated by comparing the DT with the real model and counting outputted units. In this cell, the pallets are charged in pairs. Coordination between both pallets is confirmed.

Robot Cell: A thorough analysis of processing and loading times is performed for the verification processes. The Robot Cell has also some simplifications in the operation, and therefore, waiting times are needed to achieve the desired practice. The verification of this subsystem is performed by comparing the takt times of the batched rosters when the real plant works fluidly, without waiting times in the Testing Stations.

Testing Stations: Capacities and takt times are set. Since these stations are considered black boxes, the verification step is simple and quick.

E Plant: After merging the whole model, bugs appear due to miscommunications and uncoordinated actions. A global verification is then necessary to confirm the correct performance of the system. After a fine tunning of the different stations and the addition of counters to keep track of the partial results, the entire model works as desired. To finally verify the DT, the production plan of a whole month is inputted into the model. As seen in the appendix A, the simulation lasts thirty-three days, making the DT slightly slower than the real one. This is slightly conservative, but the result is accepted. It is preferred to work from a not optimistic side to be more skeptical of the results. Throughput and productivity figures are compared to the real ones to confirm validity.

4

Simulations and Optimization of the Production Flow

This section describes all the different experiments that have been performed with the DT from a theoretical point of view. After having constructed and validated the model, it is important to use critical thinking to boost the performance of the real system. The testing of innovations in the flow is thus briefly detailed here. In the next section, all the simulation results are exposed and evaluated. Different tests were performed, and those can be divided into three large groups:

- Using the Bottleneck Analyzer in large and constant batches.
- Re-scheduling the production plant using GAs.
- Introducing changes in the production flow and evaluating those with the Experiment Manager.

As mentioned in section 3.4, the production plan used in most of the simulations is the one corresponding to the month February. However, this is different for the set of experiments that try to reach a stationary behavior. These experiments keep the same order for a long period of time. The product and the configuration are constant, and materials or tools are only changed when they are exhausted.

4.1 Bottleneck Analysis in Large and Constant Batches

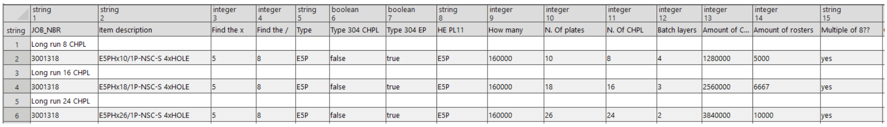
The first of the experiments performed had as a goal the identification of bottlenecks for long batches of the same product. With this strategy, a stationary behavior is reached, and therefore the results are more reliable. Three different batches were considered for the experiment, and the variable modified for those was the amount of CHPL. Eight, sixteen, and twenty-four were the figures chosen. With this choice, the testers work under different conditions and it is possible to identify the parts of the plant that cause the bottlenecks under these variable circumstances. Moreover, the OEEs of the PL and testers were also modified to obtain information about its influence. Three cases were considered:

- PLs OEE 42% Testers' OEE 100%. Real case.

- PLs OEE 100% Testers' OEE 33%. Ideal PL.

- PLs OEE 100% Testers' OEE 100%. Ideal case.

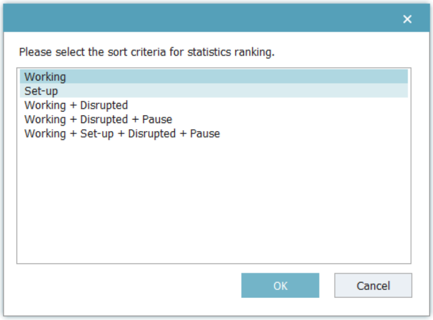
The production plan used can be seen in figure 4.1 and all the experiments were run for a maximum time of twenty days. A total of nine runs were executed.



	string 1	string 2	integer 3	integer 4	string 5	boolean 6	boolean 7	string 8	integer 9	integer 10	integer 11	integer 12	integer 13	integer 14	string 15
string	JOB_NBR	Item description	Find the x	Find the /	Type	Type 304 CHPL	Type 304 EP	HE PL11	How many	N. Of plates	N. Of CHPL	Batch layers	Amount of C...	Amount of rosters	Multiple of 8??
1	Long run 8 CHPL														
2	3001318	E5PHx10/1P-NSC-S 4xHOLE	5	8	E5P	false	true	E5P	160000	10	8	4	1280000	5000	yes
3	Long run 16 CHPL														
4	3001318	E5PHx18/1P-NSC-S 4xHOLE	5	8	E5P	false	true	E5P	160000	18	16	3	2560000	6667	yes
5	Long run 24 CHPL														
6	3001318	E5PHx26/1P-NSC-S 4xHOLE	5	8	E5P	false	true	E5P	160000	26	24	2	3840000	10000	yes

Figure 4.1: Long Runs Production Plan

The Bottleneck Analyzer can be configured to filter the results depending on the user's preferences. In this case, the chosen criterion was working percentage.



Please select the sort criteria for statistics ranking.

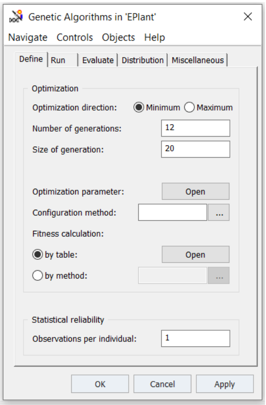
Working
Set-up
Working + Disrupted
Working + Disrupted + Pause
Working + Set-up + Disrupted + Pause

Figure 4.2: Bottleneck Analyzer Configuration

The Bottleneck Analyzer is the essential tool used in this set of tests. It is detailed in section 3.2. The generation of reports is also a fundamental utility in this experiment to find out the amount of produced units and CHPLs with varying conditions.

4.2 Optimization of the Production Plan

The second experiment performed aimed to improve the simulation time by changing the sequence of orders. The initial production plan for this simulation was a rescheduled version of the original plan for the month February. Some changes were applied to facilitate the operation of the GA Wizard. These modifications were a manual reschedule of the orders to minimize the number of tool and coil changes. In addition, all the machine's OEEs were set to 100% to make the simulation faster and to avoid variability in the results. If random parameters are used, variability in the results appears, and thus, more runs are required. In the simulation, twelve generations were chosen, and their size was set to twenty individuals (see Fig. 4.3). The total number of jobs to analyze was eighty, making the mentioned figures a reasonable trade-off between time and performance.



Optimization direction: ● Minimum ○ Maximum

Number of generations: 12

Size of generation: 20

● by table:

○ by method:

Observations per individual: 1

Figure 4.3: GA Wizard Configuration

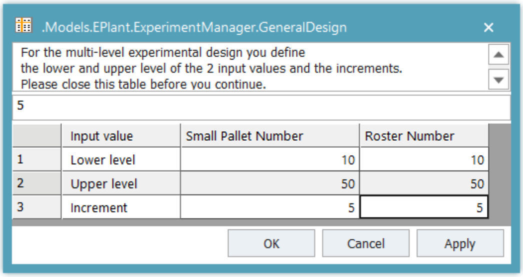
4.3 Changes in the Production Flow

In this last set of experiments, modifications in the production flow are performed. The first of the experiments modifies the number of rosters and small pallets and evaluates the performance of the production flow. The Experiment Manager is used to run all the simulations. The second modification aims to distribute the number of HEs in the testers in a balanced way. A new strategy is programmed to achieve this goal.

Experiment Manager

The Experiment Manager helps the user in executing simulation studies. It can run the same experiment with different inputs and analyze the influence of those variations. As the GA Wizard, this is somewhat more time-consuming, and simulations can last several hours.

This test aims to check the influence in the total throughput if the number of rosters and small pallets is modified. A multi-level test is then set and the main features can be seen in figures 4.4 and 4.5. The production plan is the same for all the runs. For a faster analysis, all the elements are considered ideal.



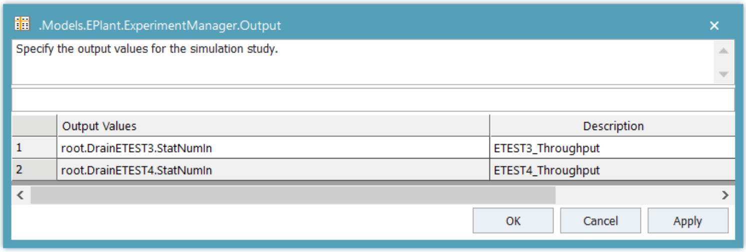
.Models.EPlant.ExperimentManager.GeneralDesign

For the multi-level experimental design you define
the lower and upper level of the 2 input values and the increments.
Please close this table before you continue.

5

	Input value	Small Pallet Number	Roster Number
1	Lower level	10	10
2	Upper level	50	50
3	Increment	5	5

Figure 4.4: Experiment Manager Input Configuration



.Models.EPlant.ExperimentManager.Output

Specify the output values for the simulation study.

	Output Values	Description
1	root.DrainETEST3.StatNumIn	ETEST3_Throughput
2	root.DrainETEST4.StatNumIn	ETEST4_Throughput

Figure 4.5: Experiment Manager Output Configuration

Smart Splitter

The last innovation tested in the DT is, as mentioned above, an innovative strategy to split the flow of HEs in a more balanced way. In figure 4.6, both methods are displayed. By only changing the target control of the robot, either the traditional method or the new one can be chosen. Again, the production plan is kept invariable.

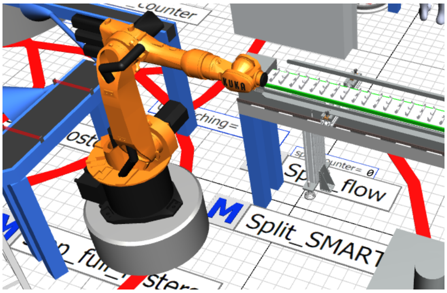


Figure 4.6: Splitter Strategy

The code of this method consists of several nested conditional statements. The main goal of this target control is to set the destination of the part that the robot grabs. If those are fixtures, they are placed immediately in their corresponding buffers. However, if the product is any kind of HE, the actual logic begins. At first, the capacities of both testers and their corresponding entrance buffers are calculated. These variables receive the names of “TEST3_NUM” and “TEST4_NUM”. Then, the robot is programmed to send the units to the station that has higher availability, in other words, these with a lower “NUM” value. In case both have the same number of parts being processed; Testing Station 4 is chosen by default. As mentioned above, the stations can be turned on and off by the user by pressing a button. This code also considers this feature. In lines nine to twelve, the code checks the availability of the testers. In case one of the stations is closed, the robot will place all the HEs in the other one. If both testers are closed, the robot will hold the HE until a station is activated again. This feature is extremely helpful to run different tests and evaluate the influence of having one or two available stations.

```
1  if @.name = "Fixture_top"
2    ?.setDestination(Conveyor23)
3  elseif @.name = "Fixture_inter"
4    ?.setDestination(Conveyor212)
5  else
6    var TEST3_NUM := conveyorEtest3_1.NumMU +
       conveyorEtest3_2.NumMU + conveyorEtest3_3.NumMU +
       conveyor4.NumMu + ETEST_3.NumMU
7    var TEST4_NUM := conveyorEtest4_1.NumMU + ETEST_4.NumMU
8
9    if ETEST4.Value = False -- ONLY ETEST3 AVAILABLE
10     ?.setDestination(ConveyorEtest3_1) --TEST3
11   elseif ETEST3.Value = False -- ONLY ETEST4 AVAILABLE
12     ?.setDestination(ConveyorEtest4_1)
13   else -- BOTH AVAILABLE
14     if conveyorEtest3_1.full
15       ?.setDestination(ConveyorEtest4_1) --TEST4
```

```
    elseif conveyorEtest4_1.full
      ?.setDestination(ConveyorEtest3_1) --TEST3
    elseif TEST4_NUM <= TEST3_NUM
      ?.setDestination(ConveyorEtest4_1) --TEST4
    else
      ?.setDestination(ConveyorEtest3_1) --TEST3
    end
  end
end
```

Listing 4.1: Smart Splitter Method

5

Results

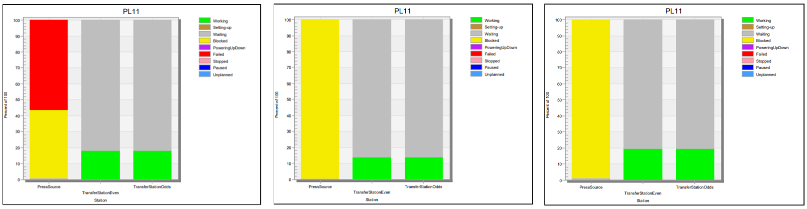
This section describes, from a practical perspective, all the results for the conducted experiments explained in the section above. A similar structure is followed and the tests are separated into three subgroups.

5.1 Bottleneck Analysis in Large and Constant Batches

The objective of this experiment is to locate the bottlenecks under different conditions. For this reason, and since the simulation time is constant in all the runs, the analysis of the changing times, material consumption, and productivity is not performed. Below, chart diagrams for all the studied cases are shown.

8 CHPLs

After analyzing the pictures below, it can be seen that the bottleneck in all the cases is the Tester Station 3, even if both testing stations are used. The furnace occupancy has different profiles, and when the DT runs in ideal mode, its occupancy will be constant between ten and twelve units. A strategy to get rid of this bottleneck would be to send more HEs to the Tester Station 4 instead of following the current policy two-one.

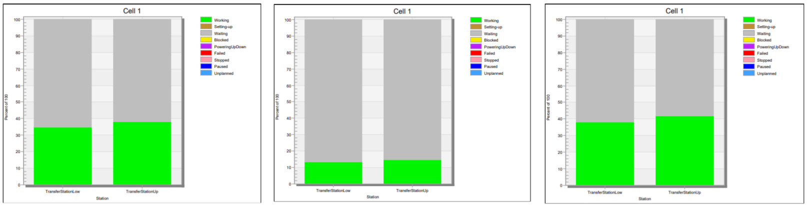


(a) Real Case

(b) Ideal PL

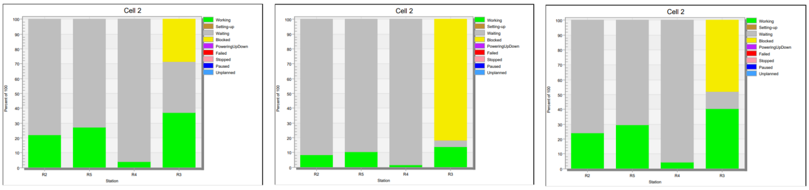
(c) Ideal Case

Figure 5.1: PL 8 CHPLs



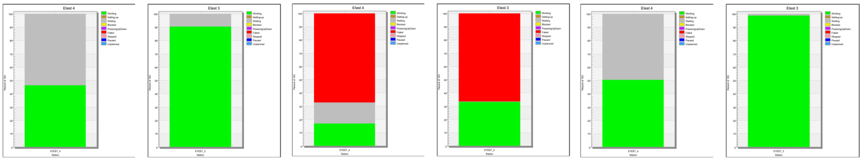
(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.2: Loading Cell 8 CHPLs



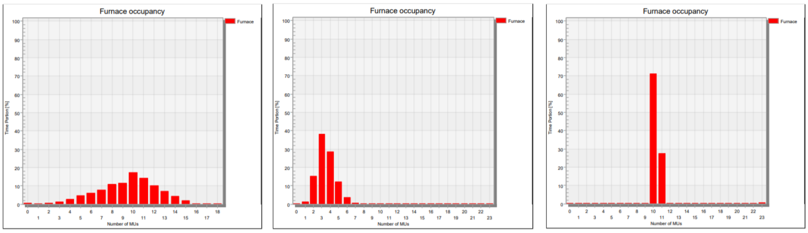
(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.3: Robot Cell 8 CHPLs



(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.4: Testers 8 CHPLs

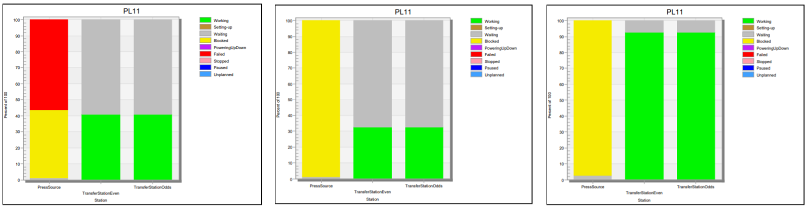


(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.5: Furnace Occupancy 8 CHPLs

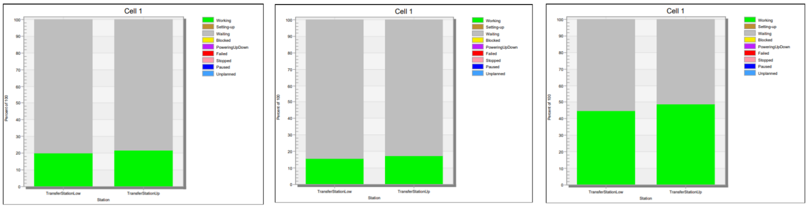
16 CHPLs

The graphs show that in this case, the PL is the bottleneck. Since the configuration is set to sixteen CHPLs, only the Tester Station 4 is working, and the Tester Station 3 does not cause the jams that are observed in the other experiments. Having the bottleneck at the beginning of the flow is adequate, due to the fact that if this operation can be done faster, the whole production speed will improve.



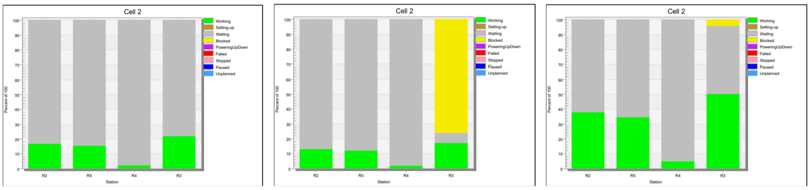
(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.6: PL 16 CHPLs



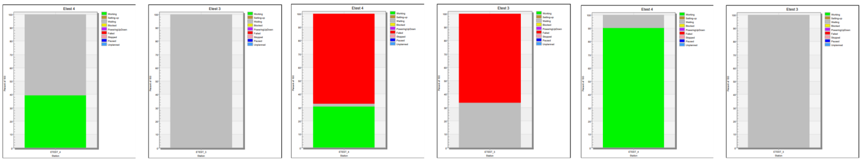
(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.7: Loading Cell 16 CHPLs



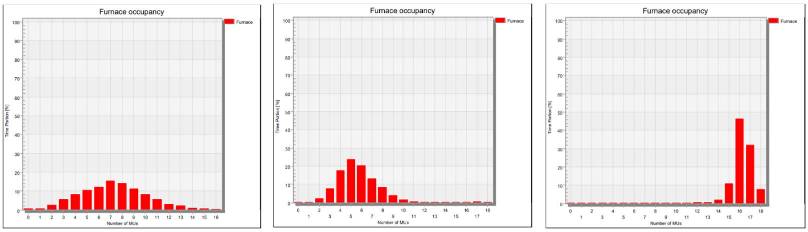
(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.8: Robot Cell 16 CHPLs



(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.9: Testers 16 CHPLs

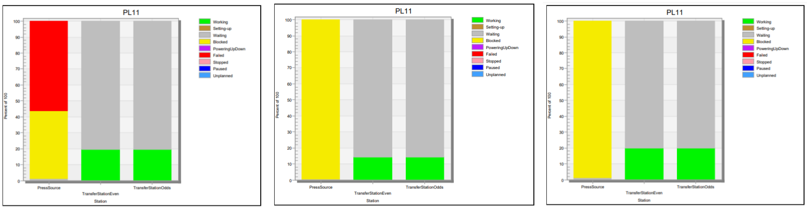


(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.10: Furnace Occupancy 16 CHPLs

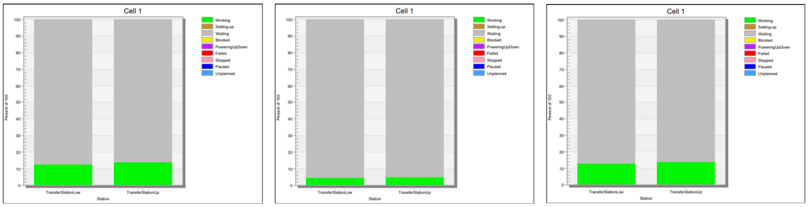
24 CHPLs

After analyzing the results, it can be observed that the bottleneck in all the cases will be Tester Station 3. Again, using the Tester Station 4 would be a good way of operating to get rid of this bottleneck.



(a) Real Case (b) Ideal PL (c) Ideal Case

Figure 5.11: PL 24 CHPLs

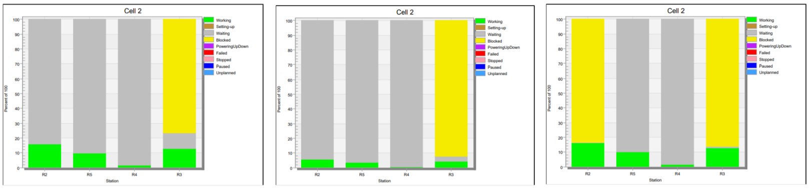


(a) Real Case

(b) Ideal PL

(c) Ideal Case

Figure 5.12: Loading Cell 24 CHPLs

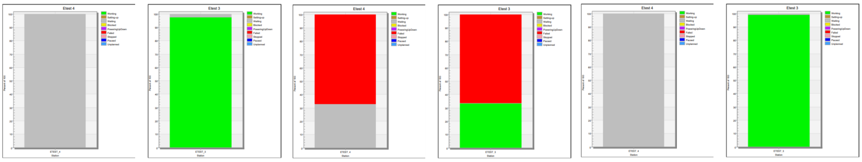


(a) Real Case

(b) Ideal PL

(c) Ideal Case

Figure 5.13: Robot Cell 24 CHPLs

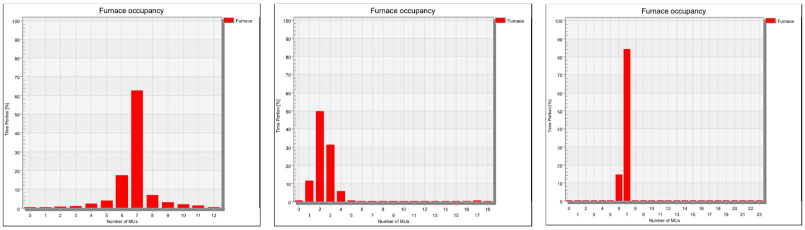


(a) Real Case

(b) Ideal PL

(c) Ideal Case

Figure 5.14: Testers 24 CHPLs



(a) Real Case

(b) Ideal PL

(c) Ideal Case

Figure 5.15: Furnace Occupancy 24 CHPLs

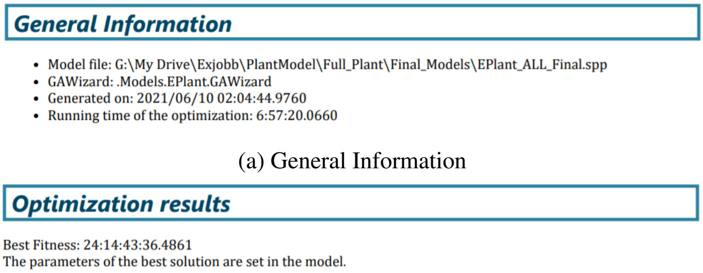
Summary:Even if the Robot Cell is probably the most complex part of the plant, it never jams the flow of products. This set of experiments justify the suggested improvements for the plant: smartly using both testers could locate the bottlenecks at the beginning of the production, as it should be ideally.

	8 CHPLs	16 CHPLs	24 CHPLs
PL 42% Testers 100%	E Test 3	PL	E Test 3
PL 100% Testers 33%	E Test 3	PL	E Test 3
PL 100% Testers 100%	E Test 3	PL	E Test 3

Table 5.1: Bottlenecks Summary

5.2 Optimization of the Production Plan

The second experiment performed includes the utilization of GAs to re-schedule the production plan. The input data for the tests is described in the section above, and the complete simulation result can be found in Appendix C. In summary, the most interesting results are gathered and discussed here.



General Information

- Model file: G:\My Drive\Exjobb\PlantModel\Full_Plant\Final_Models\EPlant_ALL_Final.spp
- GAWizard: .Models.EPlant.GAWizard
- Generated on: 2021/06/10 02:04:44.9760
- Running time of the optimization: 6:57:20.0660

(a) General Information

Optimization results

Best Fitness: 24:14:43:36.4861
The parameters of the best solution are set in the model.

(b) Optimization Results

Figure 5.16: GA Wizard Report

Experiments with GAs usually take time. In this case, the simulation lasted almost seven hours. It is important to mention that the initial production plant was modified before running the GA. Otherwise, the results could have been worse and the simulation time longer. The evolution of the results can be seen in figure 5.17. A convergence is observed towards the final result. It is also necessary to mention that optimality is not guaranteed. The result, even if it enhances the performance, may be improved with more runs or different inputs.

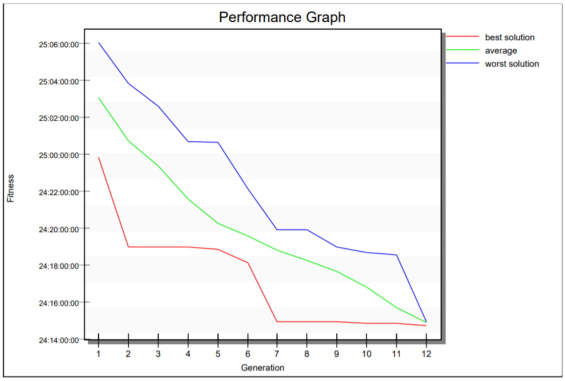
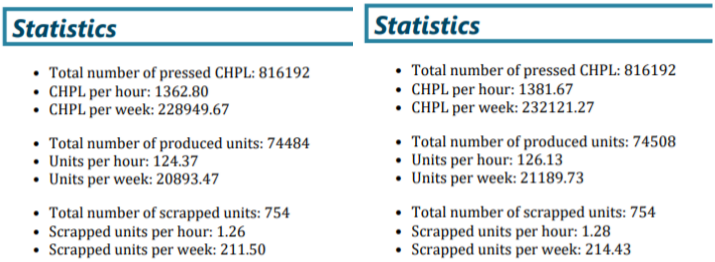


Figure 5.17: GA Wizard Generations

before and after running the GAs. In both simulations, the total number of pressed CHPLs is, logically, the same. However, after the re-scheduling process, the simulation time is around eight hours lower, changing the number of units per hour or week. If this strategy is followed for the whole year, the production could be increased by 15405 units. This is, of course, ideal and too optimistic. Urgent orders must be satisfied and at times it not possible to re-schedule some orders in particular because of lack of tools, materials, or workers. Nevertheless, this experiment demonstrates that there is room for improvement and the employment of GAs can be an interesting tool.



Statistics

- Total number of pressed CHPL: 816192
- CHPL per hour: 1362.80
- CHPL per week: 228949.67

- Total number of produced units: 74484
- Units per hour: 124.37
- Units per week: 20893.47

- Total number of scrapped units: 754
- Scrapped units per hour: 1.26
- Scrapped units per week: 211.50

(a) Before

Statistics

- Total number of pressed CHPL: 816192
- CHPL per hour: 1381.67
- CHPL per week: 232121.27

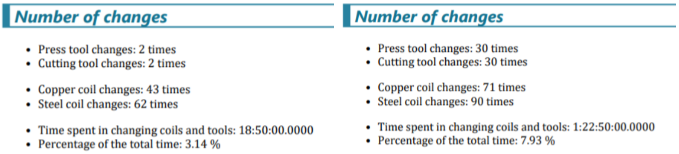
- Total number of produced units: 74508
- Units per hour: 126.13
- Units per week: 21189.73

- Total number of scrapped units: 754
- Scrapped units per hour: 1.28
- Scrapped units per week: 214.43

(b) After

Figure 5.18: Productivity Figures

As a curiosity, it can be seen in figure 5.19 that the time spent setting up has increased considerably whereas the simulation time has been decreased from 24:22:54:36.1821 to 24:14:43:36.4861 (d: h: m: s: ms). Avoiding unnecessary replacements is important,but it is even more important to balance accurately the production so that all the components work at a high rate constantly.



Number of changes

- Press tool changes: 2 times
- Cutting tool changes: 2 times

- Copper coil changes: 43 times
- Steel coil changes: 62 times

- Time spent in changing coils and tools: 18:50:00.0000
- Percentage of the total time: 3.14 %

Number of changes

- Press tool changes: 30 times
- Cutting tool changes: 30 times

- Copper coil changes: 71 times
- Steel coil changes: 90 times

- Time spent in changing coils and tools: 1:22:50:00.0000
- Percentage of the total time: 7.93 %

(a) Before

(b) After

Figure 5.19: Setup Times

5.3 Changes in the Production Flow

The last of the experiments deals with changes in the production flow. It has been detailed in the section above, and its results are discussed here.

Experiment Manager

By changing the number of rosters and small pallets we can analyze the behavior of the plant if maintenance is required and some of the elements are not available. Moreover, it can be evaluated if adding more of these elements would have a positive influence on productivity terms. A portion of the generated report can be reviewed in appendix D. The experiment is set to evaluate the influence of these inputs and can be seen in figure 4.4. Both values are changed from ten to fifty with steps of five. A total of eighty-one experiments were run, and therefore, some data caption was needed. In the graph below (Fig. 5.20) the results are displayed.

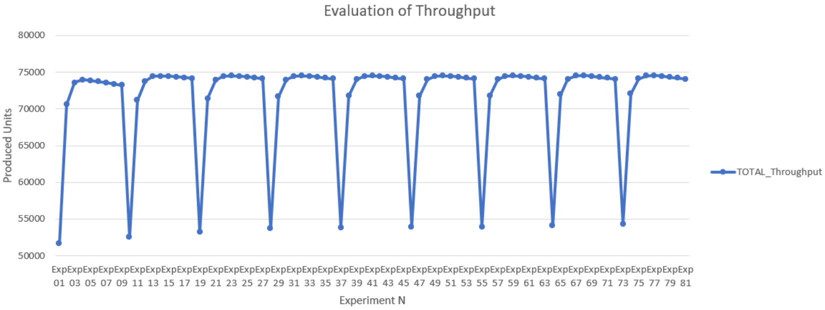


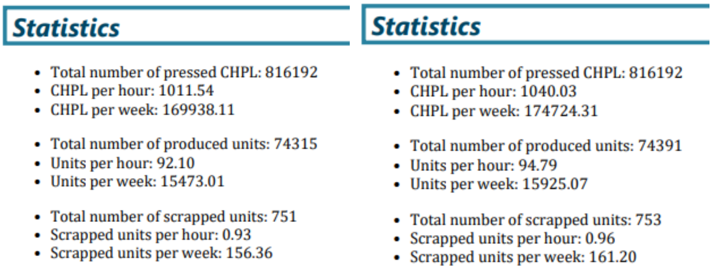
Figure 5.20: Evaluation of the Throughput with the Experiment Manager

The experiment with the highest throughput was the number 68. This experiment uses forty-five small pallets and thirty rosters. Unsurprisingly, the number is quite close to the one in reality (forty and thirty-two). The production is currently working with a quite good number of elements. According to simulations, the increase in the throughput for this example is only 2.68%.

Smart Splitter

When the method exposed in section 4.3 is set as the target control for the robot, substantial improvements are found. To give more reality, the OEE figure of the PL was set to 42%, and the orders chosen for the simulation were corresponding to the month of February.

The first figure represents the statistical outputs generated in the report for both cases, running the simulation with the old strategy or using the new one. Logically, the number of CHPL is the same since the orders are identical, and the PL stops when they are all completed. However, the number of produced units is slightly higher in the second generation. This is because the testers are more alleviated and allow the counter to reach a higher figure, and therefore the units wait less time in the conveyor in the second simulation. When checking the produced units per week, there is an increase of 2.83% if the innovative method is used. The total simulation time is decreased by a day, causing a considerable boost in productivity.



Statistics

- Total number of pressed CHPL: 816192
- CHPL per hour: 1011.54
- CHPL per week: 169938.11

- Total number of produced units: 74315
- Units per hour: 92.10
- Units per week: 15473.01

- Total number of scrapped units: 751
- Scrapped units per hour: 0.93
- Scrapped units per week: 156.36

(a) Before

Statistics

- Total number of pressed CHPL: 816192
- CHPL per hour: 1040.03
- CHPL per week: 174724.31

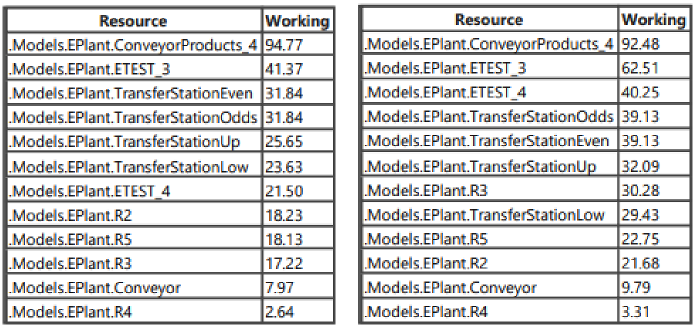
- Total number of produced units: 74391
- Units per hour: 94.79
- Units per week: 15925.07

- Total number of scrapped units: 753
- Scrapped units per hour: 0.96
- Scrapped units per week: 161.20

(b) After

Figure 5.21: Statistics Splitter Strategy

With regards to the bottlenecks, Testing Station 3 continues to be the main problem in both simulations. In both cases, the elements that belong to the Robot Cell will never cause jams, and the PL may be the next part to analyze in case the first strategy is followed.



Resource	Working
.Models.EPlant.ConveyorProducts_4	94.77
.Models.EPlant.ETEST_3	41.37
.Models.EPlant.TransferStationEven	31.84
.Models.EPlant.TransferStationOdds	31.84
.Models.EPlant.TransferStationUp	25.65
.Models.EPlant.TransferStationLow	23.63
.Models.EPlant.ETEST_4	21.50
.Models.EPlant.R2	18.23
.Models.EPlant.R5	18.13
.Models.EPlant.R3	17.22
.Models.EPlant.Conveyor	7.97
.Models.EPlant.R4	2.64

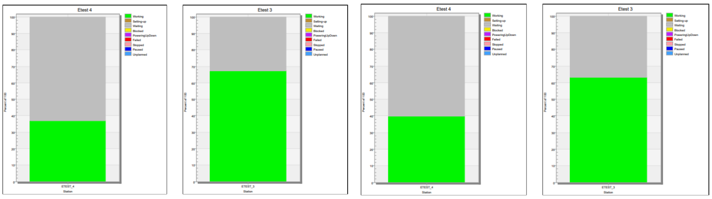
(a) Before

Resource	Working
.Models.EPlant.ConveyorProducts_4	92.48
.Models.EPlant.ETEST_3	62.51
.Models.EPlant.ETEST_4	40.25
.Models.EPlant.TransferStationOdds	39.13
.Models.EPlant.TransferStationEven	39.13
.Models.EPlant.TransferStationUp	32.09
.Models.EPlant.R3	30.28
.Models.EPlant.TransferStationLow	29.43
.Models.EPlant.R5	22.75
.Models.EPlant.R2	21.68
.Models.EPlant.Conveyor	9.79
.Models.EPlant.R4	3.31

(b) After

Figure 5.22: Bottlenecks Splitter Strategy

However, it can be seen that the testing units' utilization is much more balanced in the second experiment. Besides, this figure is also higher in this experiment than in the first one, and this means that the waiting times there are decreased and thus better overall performance is achieved.



(a) Before

(b) After

Figure 5.23: Testers Splitter Strategy

4.3

6 Discussion

This chapter wraps up all the obtained results of the project and answers the research questions that were formulated at the beginning of the project. Along with those, additional inquiries that arose during the development of the project and are also discussed here.

6.1 Creation of a Digital Twin

Creating a Digital Twin is a common practice among manufacturing companies. The transformation of modern markets demands better quality, customization of products, faster lead times, and lower costs. Manufacturers find themselves in need of improving their existing facilities by modifying the existing ones. The development of simulation and communication tools in the previous years allows this revolution.

In this project, a real simulation model is built using Tecnomatix Plant Simulation as the main software and Excel files as an auxiliary tool. Entrance barriers to using the software exist, beginner users may not access all the software's utilities, and thus models can be uncompleted or too simple. Skills are required for the successful execution of this kind of works, not only technical but also analytical. Computing skills, production technology knowledge, or other engineering fields are necessary. Yet, to model reality, planning, criticism, and analytical thinking are at least as important as the technical competencies. With this being said, in no way would I want to underestimate the labor of software engineers. Instead, I want to highlight the fact that being a flexible and open-minded person is of utmost important for this endeavor. Many inputs can be extracted from a production chain, yet filtering the most essential ones is another meaningful characteristic. Being close to the real world is important in these simulation projects. The behavior of the system is essential, but changes and unforeseen events occur, and those must be put at a superior level. Numerous variables must be taken into consideration.

Verification plays a crucial role in the digitalization of manufacturing systems. Simulations may not copy 100% of what we see in the real world, but they have to be useful and quite close to the real performances. A simulation that is not flexible and adaptable can be fruitless and counterproductive. It is not advised to use a DT until the verification is completed since it can create confusion and misunderstanding of processes and data.

6.2 Optimization of the Production Flow

Once the production is modeled, the fun begins. As an engineer, I like to test new alternatives, improve processes and create innovative systems. But let us be realistic and keep our feet on the ground. Sophisticated manufacturing systems cost millions of dollars, and we cannot change them overnight for two reasons. First, because of their cost, and second, because of their integration.

Again, critical thinking is mandatory. Which parts of the plant are feasible to be modified? In this project, two changes are evaluated: changes in the number of mobile elements and different strategies to split the flow at the end of the production. Both changes are critically evaluated and conclusions are reached. In the first one, no clear improvements are achieved. However, for long batches with the same produced HE, the experiment may be interesting. An instance could be a long batch with a low number of CHPLs. In this case, the flow of materials would be faster, and adding extra elements may help. When a high amount of CHPLs is batched, the flow of products is lower. This is the ideal moment to replace some of the rosters or bring about maintenance activities. In the second experiment, the productivity is shown to be increased by almost 3% with a small investment. Only some sensors to count parts and some adjustments in the morphology of the conveyors would make this possible. A gain in flexibility and testing capacity is also guaranteed if the modification is performed. Economical assessments would confirm the feasibility of the change and the actual amortization period.

Mathematical tools such as Genetic Algorithms or multi-level experiments are used to improve the current state of the model. These tools are used smartly, without being too optimistic and trying to be realistic. Limitations and simplifications are motivated and taken into consideration.

6.3 Professional Utility of the Project

For me, the project has been the cherry on the cake for my studies. It gave me a broad overview of how organizations work and it made me apply all the knowledge I have been gathering during my formation years.

For the company, a safe environment for testing and validating has been created and some experiments have already been performed. Nevertheless, there is still room for improvement and experimenting. Some of the limitations have been explained and in the section on future works (7.1) alternatives are presented. This section can be considered a preliminary roadmap and includes very inspiring activities that will be achieved in the near future.

7

Conclusions

In this Master's Thesis, a Digital Twin of a production flow has been created, validated, and operated to achieve better performance. At first, a brief introduction to the manufacturing world and the evolution of the industry motivated the choice of the project. The theoretical aspects required for the whole understanding of the project are then explained. After this point, the actual work begins. The process of constructing and validating the system and its parts is detailed. Once the validation process is completed, three different groups of experiments are exposed, simulated, and evaluated.

The first conclusion that can be drawn is that it is possible to build a realistic model using TPS. This powerful software has a very intuitive GUI that can be understood quickly. However, when sophisticated models are created, coding skills are required. TPS has a small variety of blocks and functions, but if the user is skilled, any complex model or system can be created by coding. It provides a wide variety of optimization and experimentation tools that can be used to increase productivity or throughputs.

After constructing the Digital Twin, a validation process was carried out. Real and specific orders were inputted into the model to evaluate its outputs and compare them with the real facility.

Once the model worked as desired, experiments were prepared, simulated, and evaluated. A total of three sets of experiments were conducted. The first one consisted of the location of bottlenecks for constant batches. The second one aimed to modify the production plan by re-scheduling orders to achieve a faster completion. The last one included two experiments that targeted increase productivity by changing parameters of the plant. A multi-level experiment and an innovative solution to balance the flow of products were tested.

The results obtained by experimentation suggested that improvements in the plant may be achieved but further research is necessary. Nevertheless, this DT is, without any doubt, a useful tool that can be used to schedule the orders and to simulate modifications in the plant.

7.1 Future Works

Even though the objectives of the project have been accomplished, there is room for improvement and further research. What is more, after creating this Digital Twin, I acquired skills that allow me to create more complex models, and to think about the potential improvements of this system in particular. With this, a roadmap with the following steps has been formulated.

The model contains only the fully automated part of the facility. Thus, the first, and very ambitious future work would be to add the remaining part. With this, workers, workplaces, and Automated Guided Vehicles (AGVs) will be added to the DT. After this process, a complete model of the testing stations will be created. As mentioned, they are considered as black boxes in the simulation, only considering the capacities and takt times. The output of the testers may be valid or invalid. Now, only scrapped units are considered, however, some of the failed units are rebrazed in a different furnace and tested again. Rebrazing processes will be considered in the future. Another step would be to standardize the creation of some of the parts of the system (Press Lines or Testing Stations for example) to use them in different models. Finally, visual improvements of the model its Mobile Units will be done to give a better visualization.

After performing these changes, another testing process will commence. This time, more radical changes will be performed and changes in the layout will be carried out. With this safe environment for testing, ambitious modifications will be constructed and evaluated economically.

Bibliography

[1] Angira Sharma, Edward Kosasih, Jie Zhang, Alexandra Brintrup, Anisoara Calinescu. “Digital twins: state of the art theory and practice, challenges, and open research questions”. *Research Gate* (2020).

[2] Fei Tao, He Zang, Ang Liu, A. Y. C. Nee. “Digital twin in industry: state-of-the-art”. *IEEE Transactions on Industrial Informatics* **15**:4 (2019).

[3] Flávia Pires, Ana Cachada, José Barbosa, António Paulo Moreira, Paulo Leitão. “Digital twin in industry 4.0: technologies, applications and challenges”. *IEEE 17th International Conference on Industrial Informatics (INDIN)* **1** (2019), pp. 721–726. DOI: 10.1109/INDIN41052.2019.8972134.

[4] Guodong Shao, Sanjay Jain, Christoph Laroque, Loo Hay Lee, Peter Lendermann, Oliver Rose. “Digital twin for smart manufacturing: the simulation aspect”. *Proceedings of the 2019 Winter Simulation Conference* ().

[5] Hitesh Hinduja, Shreya Kekkar, Smruti Chourasia, Hrishkedsh Bharadwaj Chakrapani. “Industry 4.0: digital twin and its industrial applications”. *International Journal of Science, Engineering and Technology* (2020).

[6] Laguna Manuel, Marklund Johan. *Business Process Modeling, Simulation and Design (3rd edn)*. Taylor & Francis Group, 2019.

[7] Law, A. M. *Simulation Modeling and Analysis (5th edn)*. McGraw-Hil, 2015.

[8] Marko Starbek, Darko Menart. “The optimization of material flow in production”. *International Journal of Machine Tools and Manufacture* 40 (2000), pp. 1299–1310.

[9] Martijn R.K. Mes. *Simulation Modelling using Practical Examples: A Plant Simulation Tutorial*. University of Twente, 2017.

[10] Robinson, S. *Simulation: The Practice of Model Development and Use (2nd edn)*. Palgrave Macmillan, 2014.

[11] Rosen R, von Wichert G, Lo G, Bettenhausen KD. “About the importance of autonomy and digital twins for the future of manufacturing”. *IFAC-PapersOnLine* 48 (2015), pp. 567–572.

[12] Siderska, Julia. “Application of tecnomatix plant simulation for modeling production and logistic processes”. *Business, Management and Education* **14** (2016), pp. 64–73. DOI: `10.3846/bme.2016.316`.

[13] SWEP. URL: `https://www.swep.net/`. (accessed: 01.06.2020).

[14] Tuegel, E.J., Ingraffea, A.R., Eason, T.G., Spottswood, S.M. “Reengineering aircraft structural life prediction using a digital twin”. *International Journal of Aerospace Engineering* **14** (2011). DOI: `10.1155/2011/154798.`.

A

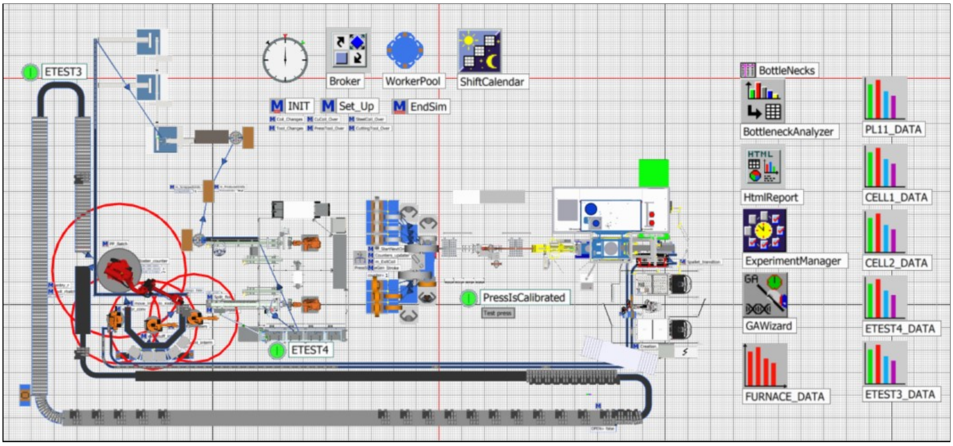
Report February

General Information

Overview

- Model file: G:\My Drive\Exjobb\PlantModel\Full_Plant\Final_Models\EPlant_ALL_Final.spp
- Simulation root: .Models.EPlant
- Simulation time: 33:14:53:00.5078
- Number of completed jobs: 81

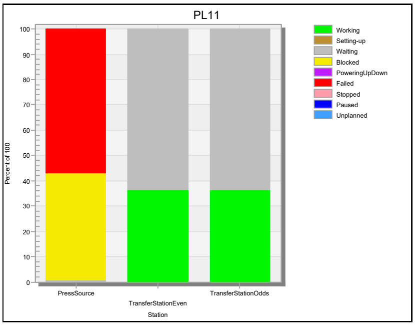
Model

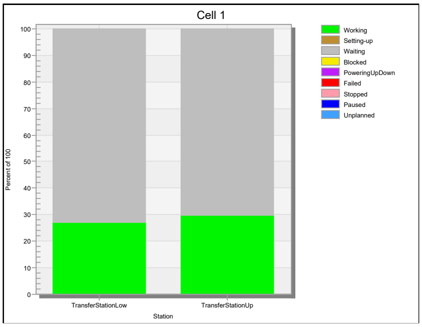


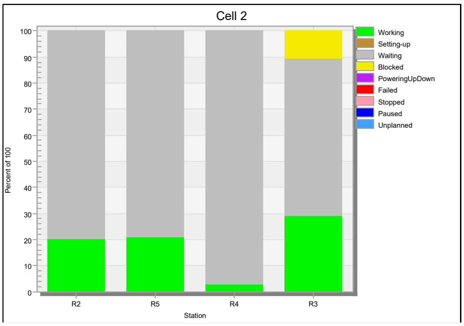
EPlant

Simulation results

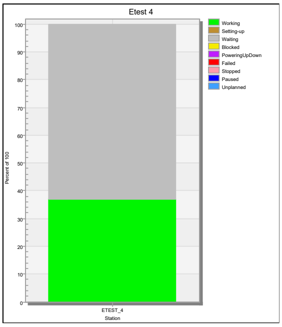
PL11, Cell1 and Cell 2

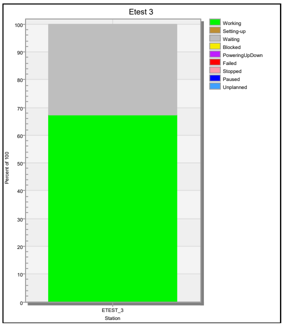




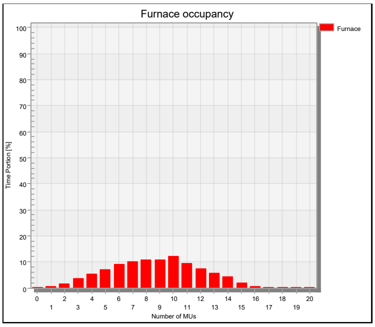


Tester Stations





Furnace occupancy



Statistics

- Total number of pressed CHPL: 816192
- CHPL per hour: 1011.54
- CHPL per week: 169938.11

- Total number of produced units: 74315
- Units per hour: 92.10
- Units per week: 15473.01

- Total number of scrapped units: 751
- Scrapped units per hour: 0.93
- Scrapped units per week: 156.36

Drain Statistics

Number of products produced in the continuous flow:

Name	Mean Life Time	Throughput	Throughput per Hour	Production	Transport	Storage	Value added	Portion
E5AS	9:10:05.4319	6621	8.21	0.64%	99.36%	0.00%	0.50%	🟨
E5P	8:41:47.7018	17026	21.10	0.57%	99.43%	0.00%	0.52%	🟨

.Models.EPlant.DrainETEST3

Name	Mean Life Time	Throughput	Throughput per Hour	Production	Transport	Storage	Value added	Portion
E5AS	8:57:07.6018	12591	15.60	0.55%	99.45%	0.00%	0.26%	🟨
E5P	8:20:41.5704	38077	47.19	0.44%	99.56%	0.00%	0.28%	🟨

.Models.EPlant.DrainETEST4

Name	Mean Life Time	Throughput	Throughput per Hour	Production	Transport	Storage	Value added	Portion
E5AS	9:03:22.3125	196	0.24	0.71%	99.29%	0.00%	0.34%	🟨
E5P	8:30:00.9650	555	0.69	0.75%	99.25%	0.00%	0.36%	🟨

.Models.EPlant.DrainSCRAPPED

Materials utilization

- Copper Coil 40434: 30 units
- Copper Coil 41597: 13 units
- Copper Coil 41544: 1 units

- Steel Coil 40126: 39 units
- Steel Coil 40001: 23 units
- Steel Coil 41610: 1 units
- Steel Coil 41611: 1 units

Number of changes

- Press tool changes: 28 times
- Cutting tool changes: 28 times

- Copper coil changes: 69 times
- Steel coil changes: 88 times

- Time spent in changing coils and tools: 1:20:50:00.0000
- Percentage of the total time: 5.56 %

BottleNeck analysis

Resource	Working
.Models.EPlant.ETEST_3	71.00
.Models.EPlant.TransferStationOdds	37.36
.Models.EPlant.TransferStationEven	37.36
.Models.EPlant.ETEST_4	35.97
.Models.EPlant.TransferStationUp	30.17
.Models.EPlant.R3	28.91
.Models.EPlant.TransferStationLow	27.67
.Models.EPlant.R5	21.39
.Models.EPlant.R2	20.70
.Models.EPlant.Conveyor	9.34
.Models.EPlant.R4	3.12
.Models.EPlant.SourceCu40434	0.00

Landskrona, Sweden 13.6.2021
Sergio Mora Carrión

B

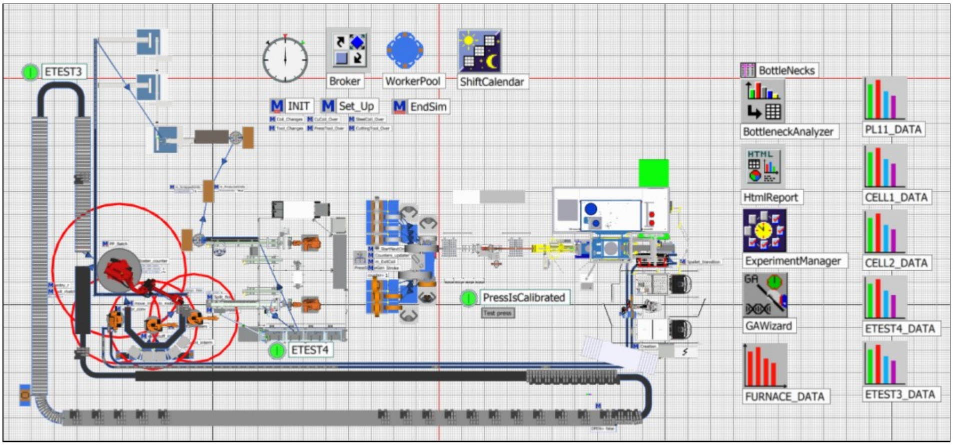
Report Long Run With 16 CHPLs

General Information

Overview

- Model file: G:\My Drive\Exjobb\PlantModel\Full_Plant\Final_Models\EPlant_ALL_Final.spp
- Simulation root: .Models.EPlant
- Simulation time: 20:00:00:00.0000
- Number of completed jobs: 4

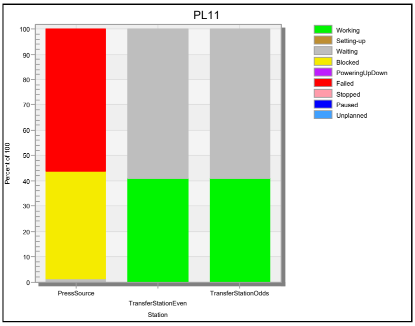
Model

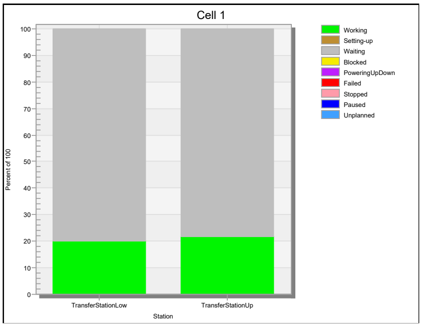


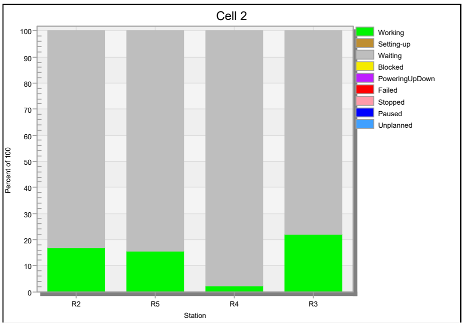
EPlant

Simulation results

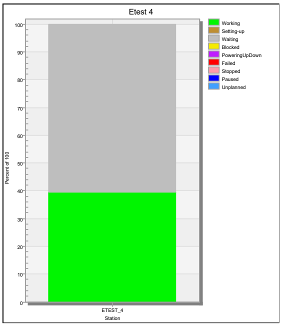
PL11, Cell1 and Cell 2

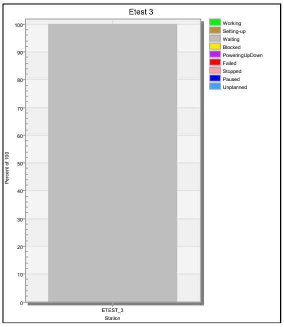




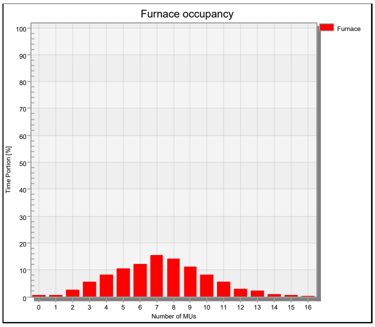


Tester Stations





Furnace occupancy



Statistics

- Total number of pressed CHPL: 532852
- CHPL per hour: 1110.11
- CHPL per week: 186498.20

- Total number of produced units: 32289
- Units per hour: 67.27
- Units per week: 11301.15

- Total number of scrapped units: 327
- Scrapped units per hour: 0.68
- Scrapped units per week: 114.45

Drain Statistics

Number of products produced in the continuous flow:

Name	Mean Life Time	Throughput	Throughput per Hour	Production	Transport	Storage	Value added	Portion

.Models.EPlant.DrainETEST3

Name	Mean Life Time	Throughput	Throughput per Hour	Production	Transport	Storage	Value added	Portion
E5P	9:55:58.9282	32289	67.27	0.24%	99.76%	0.00%	0.23%	🟧

.Models.EPlant.DrainETEST4

Name	Mean Life Time	Throughput	Throughput per Hour	Production	Transport	Storage	Value added	Portion
E5P	9:56:07.7468	327	0.68	0.25%	99.75%	0.00%	0.23%	🟧

.Models.EPlant.DrainSCRAPPED

Materials utilization

- Copper Coil 40434: 28 units
- Copper Coil 41597: 1 units
- Copper Coil 41544: 1 units

- Steel Coil 40126: 36 units
- Steel Coil 40001: 1 units
- Steel Coil 41610: 1 units
- Steel Coil 41611: 1 units

Number of changes

- Press tool changes: 1 times
- Cutting tool changes: 1 times

- Copper coil changes: 27 times
- Steel coil changes: 35 times

- Time spent in changing coils and tools: 11:00:00.0000
- Percentage of the total time: 2.29 %

BottleNeck analysis

Resource	Working
.Models.EPlant.ConveyorProducts_4	95.33
.Models.EPlant.TransferStationOdds	42.59
.Models.EPlant.TransferStationEven	42.59
.Models.EPlant.ETEST_4	40.24
.Models.EPlant.TransferStationUp	22.52
.Models.EPlant.R3	22.50
.Models.EPlant.TransferStationLow	20.65
.Models.EPlant.R2	17.50
.Models.EPlant.R5	15.97
.Models.EPlant.Conveyor	10.65
.Models.EPlant.R4	2.33
.Models.EPlant.SourceCu40434	0.00

Landskrona, Sweden 13.6.2021
Sergio Mora Carrión

C

GA Report

SIEMENS

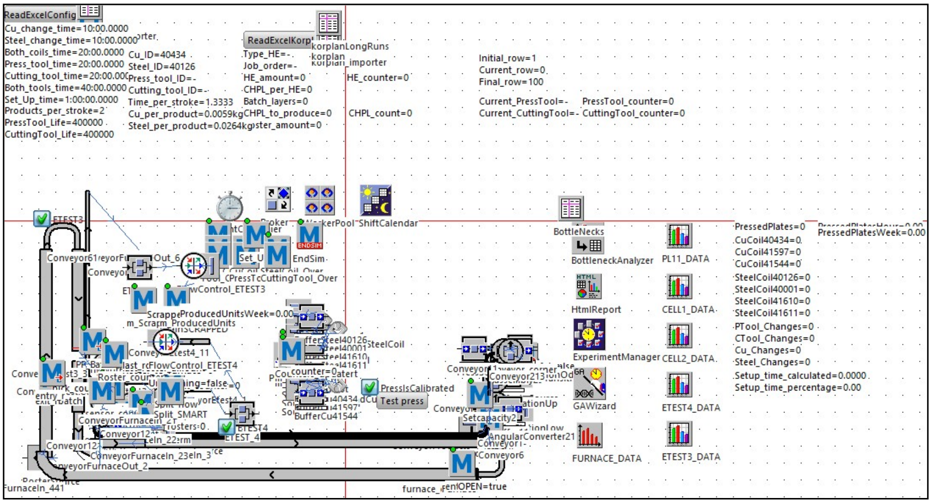
Tecnomatix

Manufacturing Simulation and Validation

General Information

- Model file: G:\My Drive\Exjobb\PlantModel\Full_Plant\Final_Models\EPlant_ALL_Final.spp
- GAWizard: .Models.EPlant.GAWizard
- Generated on: 2021/06/10 02:04:44.9760
- Running time of the optimization: 6:57:20.0660

Model



.Models.EPlant

Optimization results

Best Fitness: 24:14:43:36.4861
The parameters of the best solution are set in the model.

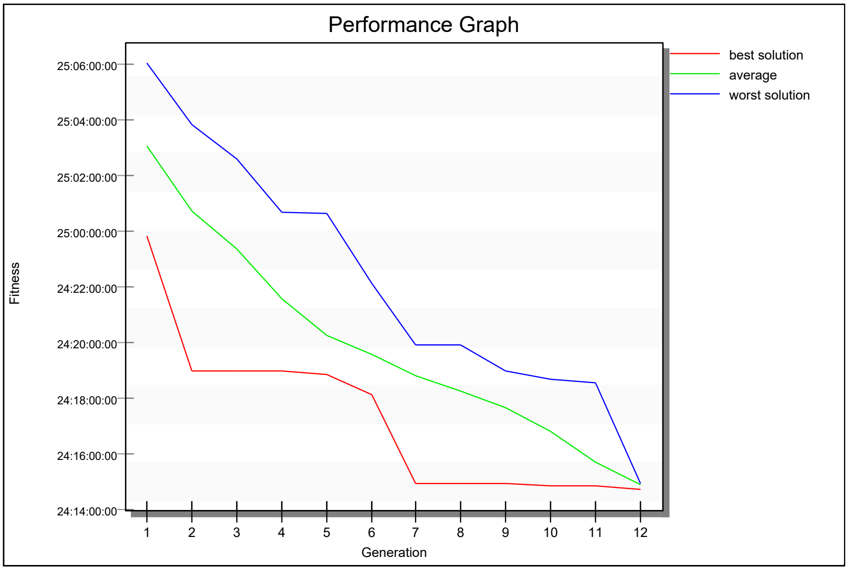
Fitness calculation

root.eventcontroller.simtime with weighting 1

Best solutions of the sequence problems

.Models.EPlant.korplan
56, 12, 75, 65, 50, 21, 25, 36, 49, 14, 33, 41, 46, 57, 70, 8, 43, 28, 73, 79, 42, 32, 61, 13, 20, 34, 27, 68, 24, 15, 2, 18, 69, 52, 67, 7, 44, 38, 19, 45, 58, 60, 30, 29, 66, 80, 40, 6, 26, 1, 4, 48, 53, 11, 47, 23, 77, 72, 76, 16, 5, 71, 3, 31, 22, 39, 35, 51, 10, 78, 17, 74, 59, 63, 37, 54, 55, 64, 9, 62

Evolution of the fitness values of the generations



Settings

Definition of optimization parameter

Parameter:	root.korplan
Sequence of	root.korplan
80 Elements	

Settings of the Genetic Algorithm

Direction of the Optimization: Minimum
Number of Generations: 12
Size of Generation: 20
Observations per individual: 1

Generated individuals

The Genetic Algorithm generated 460 individuals.
The search for equal individuals is performed.
Number of multiple generated individuals: 98
The penalty method was not applied.
Number of evaluated individuals: 362
Observations per individual: 1
362 simulation runs are performed.

D

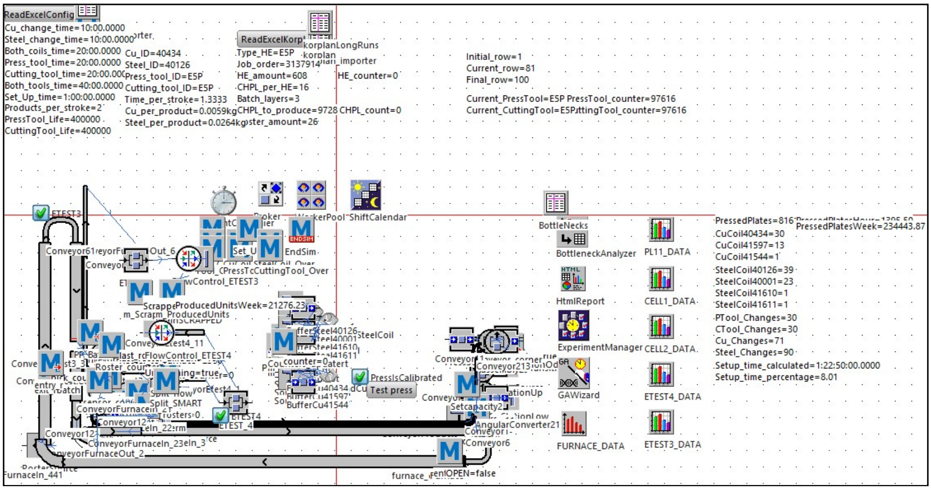
Report Experiment Manager

SIEMENS Tecnomatix
Manufacturing Simulation and Validation

General Information

- Model file: G:\My Drive\Exjobb\PlantModel\Full_Plant\Final_Models\EPlant_ALL_Final.spp
- ExperimentManager: .Models.EPlant.ExperimentManager
- Generated on: 2021/06/10 22:46:25.6380
- The running time was 1:25:45.4230.

Model



```
Cu_change_time=10:00.0000
Steel_change_time=10:00.0000
Both_coils_time=20:00.0000
Press_tool_time=20:00.0000
Cutting_tool_time=20:00.000
Both_tools_time=40:00.0000
Set_Up_time=1:00:00.0000
Products_per_stroke=2
PressTool_Life=400000
CuttingTool_Life=400000

Cu_ID=40434
Steel_ID=40126
Press_tool_ID=E5P
Cutting_tool_ID=E5P
Time_per_stroke=1.3333
Cu_per_product=0.0059kg
Steel_per_product=0.0264kg

Type_HE=E5P
Job_order=3137914
HE_amount=608
CHPL_per_HE=16
Batch_layers=3
CHPL_to_produce=9728
HE_counter=0
CHPL_count=0

Initial_row=1
Current_row=81
Final_row=100

Current_PressTool=E5P PressTool_counter=97616
Current_CuttingTool=E5P CuttingTool_counter=97616

PressedPlates=816
PressedPlatesWeek=234443.87
CuCoil40434=30
CuCoil41597=13
CuCoil41544=1
SteelCoil40126=39
SteelCoil40001=23
SteelCoil41610=1
SteelCoil41611=1
PTool_Changes=30
CTool_Changes=30
Cu_Changes=71
Steel_Changes=90
Setup_time_calculated=1:22:50:00.0000
Setup_time_percentage=8.01

ScrappedProducedUnitsWeek=21276.23
furnace_OpenOPEN=false
```

.Models.EPlant

Overview

Overview of all executed experiments, their parametrizations and the mean values of the target values.
The number of decimal places of the results is -1.

	Small Pallet Number	Roster Number	ETEST3_Throughput	ETEST4_Throughput
Exp 01	10	10	17319	34323
Exp 02	10	15	23550	47060
Exp 03	10	20	24506	49068
Exp 04	10	25	24512	49440
Exp 05	10	30	24506	49337
Exp 06	10	35	24506	49195
Exp 07	10	40	24506	49040
Exp 08	10	45	24468	48898
Exp 09	10	50	24417	48795
Exp 10	15	10	17616	34923
Exp 11	15	15	23734	47428
Exp 12	15	20	24506	49242
Exp 13	15	25	24722	49705
Exp 14	15	30	24738	49705
Exp 15	15	35	24726	49705
Exp 16	15	40	24635	49705
Exp 17	15	45	24585	49642
Exp 18	15	50	24553	49579
Exp 19	20	10	17847	35386
Exp 20	20	15	23811	47591
Exp 21	20	20	24504	49401
Exp 22	20	25	24751	49709
Exp 23	20	30	24795	49709
Exp 24	20	35	24751	49709
Exp 25	20	40	24633	49709
Exp 26	20	45	24583	49642
Exp 27	20	50	24546	49559
Exp 28	25	10	17986	35695
Exp 29	25	15	23902	47769
Exp 30	25	20	24518	49460
Exp 31	25	25	24759	49705
Exp 32	25	30	24801	49705
Exp 33	25	35	24746	49705
Exp 34	25	40	24625	49709
Exp 35	25	45	24583	49638
Exp 36	25	50	24544	49547
Exp 37	30	10	18051	35786
Exp 38	30	15	23928	47817
Exp 39	30	20	24524	49472
Exp 40	30	25	24761	49705
Exp 41	30	30	24805	49705
Exp 42	30	35	24746	49705
Exp 43	30	40	24627	49705
Exp 44	30	45	24583	49635
Exp 45	30	50	24544	49543
Exp 46	35	10	18083	35877
Exp 47	35	15	23924	47817
Exp 48	35	20	24526	49484
Exp 49	35	25	24757	49709
Exp 50	35	30	24795	49705
Exp 51	35	35	24732	49705
Exp 52	35	40	24621	49705
Exp 53	35	45	24579	49630
Exp 54	35	50	24540	49543
Exp 55	40	10	18075	35857
Exp 56	40	15	23930	47824
Exp 57	40	20	24534	49484
Exp 58	40	25	24761	49705

	Small Pallet Number	Roster Number	ETEST3_Throughput	ETEST4_Throughput
Exp 59	40	30	24795	49709
Exp 60	40	35	24732	49705
Exp 61	40	40	24619	49705
Exp 62	40	45	24579	49630
Exp 63	40	50	24542	49539
Exp 64	45	10	18134	35964
Exp 65	45	15	23991	47947
Exp 66	45	20	24548	49519
Exp 67	45	25	24781	49709
Exp 68	45	30	24819	49709
Exp 69	45	35	24722	49705
Exp 70	45	40	24615	49701
Exp 71	45	45	24575	49626
Exp 72	45	50	24538	49535
Exp 73	50	10	18202	36111
Exp 74	50	15	24029	48018
Exp 75	50	20	24553	49531
Exp 76	50	25	24779	49705
Exp 77	50	30	24813	49705
Exp 78	50	35	24718	49705
Exp 79	50	40	24615	49694
Exp 80	50	45	24575	49622
Exp 81	50	50	24540	49531

Simulation effort: 81 experiments with 81 simulation runs
No special diagrams

Values of experiments

Input values (Model parameters)

Input values	set by	Original Value	Inherited	Technical Notation
Small Pallet Number	Tab			root.Source.Number
Roster Number	Tab			root.RosterSource.Number

The input values are set by either the Table **ExpTable** or by actions of rules.

Output values (Results of the simulation study)

Target value	evaluated by	Technical Notation	High Values are Better
ETEST3_Throughput	Tab	root.DrainETEST3.StatNumIn	
ETEST4_Throughput	Tab	root.DrainETEST4.StatNumIn	

The Output values are evaluated by the table **DetailedResults** or by conditions of rules.

Multi-level experimental design

Design table of the experiment study

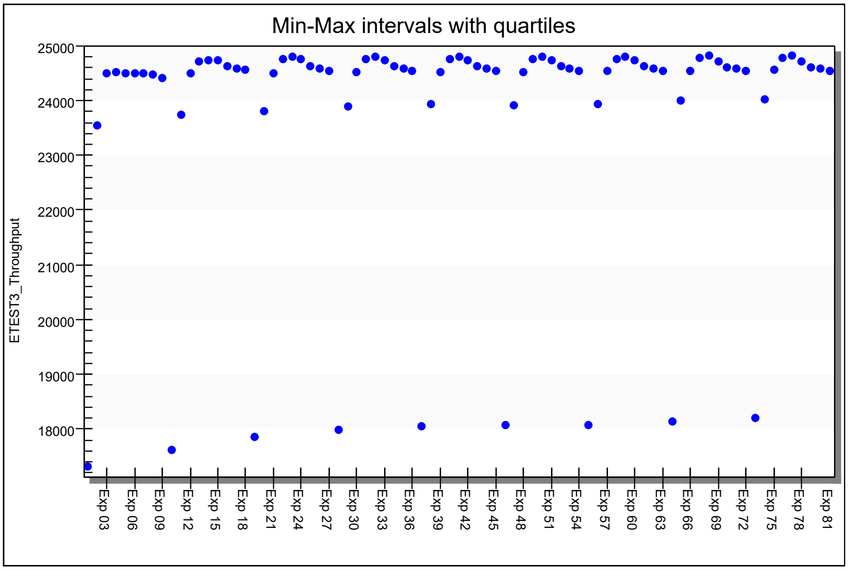
Input value	Small Pallet Number	Roster Number
Lower level	10	10
Upper level	50	50
Increment	5	5

Statistical Evaluations

- Statistical reliability
  Observations per experiment: 1
  Confidence level (%): 95

Statistics of output values

Evaluations of the output value 'ETEST3_Throughput'


Min-Max intervals with quartiles
ETEST3_Throughput
Exp 03
Exp 06
Exp 09
Exp 12
Exp 15
Exp 18
Exp 21
Exp 24
Exp 27
Exp 30
Exp 33
Exp 36
Exp 39
Exp 42
Exp 45
Exp 48
Exp 51
Exp 54
Exp 57
Exp 60
Exp 63
Exp 66
Exp 69
Exp 72
Exp 75
Exp 78
Exp 81