Bachelor Degree Project

UNIVERSITY OF SKÖVDE

EVALUATION OF VIRTUAL CONFIDENCE: TURN-TABLE MODEL & MECHATRONIC CONCEPT DESIGNER

Bachelor Degree Project in Mechanical Engineering
30 ECTS
Spring term 2018

Authors: Xabier Gastañares & Iñaki Bastida

Supervisor: Jan Orcarsson

Examiner: Anders Biel

Abstract

This work presents an evaluation of the virtual confidence that can be achieved in the software Mechatronic Concept Designer. The virtual confidence is defined as the reliability that a model or software offer, based on their capacity to replicate reality accurately. Reference system for the evaluation is a turn table used at AB Volvo in the production of front lids. After conducting the simulations and different kind of analyses (such as friction analyses), the main conclusion has been that both the original model and the software offered some limitations that made a high level of virtual confidence impossible. Taking into account this fact, future improvement lines have been suggested.

Certification

This thesis has been submitted by Xabier Gastañares and Iñaki Bastida to the University of Skövde as a requirement for the degree of Bachelor of Science in Mechanical Engineering. The undersigned certifies that all the material in this thesis that is not my own has been properly acknowledged using accepted referencing practices and, further, that the thesis includes no material for which I have previously received academic credit.

Xabier Gastañares   Iñaki Bastida

Skövde 2018-05-22

Institutionen för Ingenjörsvetenskap/Department of Engineering Science

Table of Contents

List of Figures

List of Tables

1. Introduction

The introduction is a summary of the background, purpose, and significance of the study.

1.1 Background

As stated by (Oscarsson et. al. 2016) "the design of new products requires numerous experiments and test-runs of new facilities that delays the product release and causes high costs if performed in the real world. Virtualization promises to reduce these costs by simulating the reality. However, the results of the simulation must predict the real results to be useful. This is called virtual confidence."

They also stated that there are five different grades of virtual confidence. This is the background that leads to this project, in which the virtual confidence of a mechatronic system model will be analysed.

1.2 Problem Statement

According to literature and vendor information mechatronic devices can be simulated with a high level of detail including mechanical-, electrical- and control properties. This project aims at investigating the possibilities provided by mechatronic system simulators. In this respect is a mechatronic system defined as an artefact which includes the previously mentioned properties (Bishop 2007), but in this project restricted to manufacturing devices.

The problem is to understand which level of virtual confidence can be achieved in a determined virtual model of a mechatronic device. This study is based on an artefact used by AB Volvo at its Umeå plant and the related requirements on input data and analysis features. This is to be divided in the three main domains of mechanical-, electrical- and control separately and combined, i.e. systems engineering aspects.

The general outcome shall be an understanding and a statement regarding the possibilities to use mechatronic systems simulations for e.g. virtual commissioning purposes (specifically for the given device). The main limitation is that the project will not specify a "real" mechatronic device. A model system will be used for evaluation purposes.

The outcome of this simulation related project is relevant for the whole community of engineering, as nowadays time and cost efficiency of product development processes is becoming a must. Simulations with a high level of virtual confidence can help developing better products cheaper and faster while decreasing all the pollution that tests might cause.

This project is intended to help making the future companies more economically, socially and ecologically sustainable, which at the same time will improve the life quality of our society.

1.3 Objectives

The main goal is to achieve an understanding and create a memory of the virtual confidence that can be achieved with a model of a manufacturing mechatronic device.

- Gain a theoretical understanding about mechatronic simulation
- Identify different simulation technologies or systems for mechatronic simulation
- For a given mechatronic simulator and model evaluate its possibilities to simulate real conditions and determine which level of virtual confidence can be achieved
- With respect to the previous goal, understand the requirements on input data and analysis features

1.4 Overview of the thesis

The thesis is divided in two main parts: the literature review and the simulation analyses using the software. The report follows this logical order; first, the frame of reference and literature review are presented, which will serve as an information background for the reader. Then, the main part of the project follows, which is how the simulations were prepared, the problems faced and the results. Finally, the project is auto-evaluated in the discussion and the conclusions are presented.

2. Frame of reference & literature review

Relevant concepts and previous works about them such as books, dissertations, conference papers and case studies have been analysed and are presented in this section.

2.1 Mechatronic systems

As stated by Bishop "the word, mechatronics, is composed of "mecha" from mechanics and the "tronics" from electronics. It is the synergetic integration of mechanical engineering, with electronics and intelligent computer control, thinking in the design and manufacturing of industrial products and processes (figure 1).
Mechatronics is the application of complex decision making to the operation of physical systems. It is a methodology used for the optimal design of electromechanical products" (Bishop 2007).

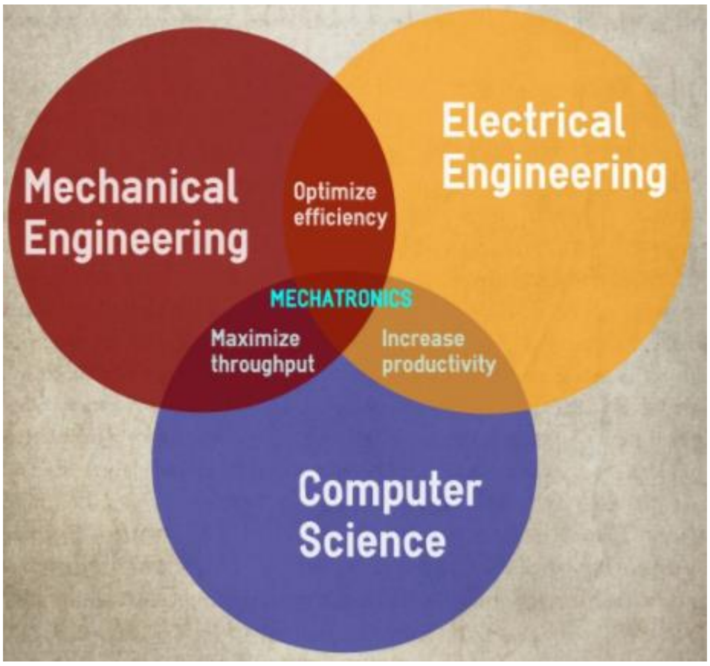

Figure 1: How mechatronics is composed (retrieved from Bishop 2007, p. 5)

Mechatronic systems have become really important over time and it is a fact that mechatronics is evolutionary. Mechatronic systems provide us many advantages (Just Science 2017) which have been shown in the following points:

- Because of the intelligence of the design of mechatronic systems, the efficiency of these systems increases.

- They are cheaper than the traditional mechanical solutions.
- The functionality and characteristics of the systems improve with mechatronics.
- Some features of products are enhanced such as the size, reliability and the design time.
- Mechatronic systems are easier to control and also safer.
- Variables like precision, position, speed and flow rate are possible to control by the microcontrollers.

2.1.1 Applications

Mechatronics systems are used in a lot of areas such as:

- Automotive: Mechatronics systems are used in automobiles for the car engine management, electronic stability control, anti-lock braking system, etc.
- Aerospace: Mechatronics is used in aeronautics, for example, for automatic pilots or aerial vehicles without crew.
- Consumer products: Mechatronics systems can usually be seen in home appliances. An example is the washing machine.
- Defence systems: These systems can be used for saving peoples life. For example, if a flood happens somewhere, people can be detected with mechatronic systems. Defence systems are also used for military applications.
- Manufacturing: Mechatronics can be found in so much manufacturing appliances such as production line automation, measuring devices, control systems, etc.
- Medical: The use of mechatronics in this area is extensive. It can be used from positioning systems in hospital beds to robotic surgical devices.
- Etc.

2.1.2 Elements of mechatronics

Generally, mechatronic systems contain these different elements (Elements of Mechatronics 2010):

- **Mechanical:** These elements are the hydraulic, thermo-fluid, mechanism and mechanical structure aspects of a mechatronic system. The elements interact with the environment and must contain static/dynamic characteristics. They need a source of energy in order to create forces, motions, etc.

- **Electro-mechanical:** This area is formed by the actuators and sensors. Sensors are used to measure variables such as light, displacement and level, touch, temperature, stress, etc. Then, actuators as servomotors, pumps, etc. have the function to activate the mechanical elements.
- **Electrical-electronic:** In this area, on the one hand, they can be found electrical elements which are analogue signals, circuits and electrical components (capacitor, transformer, etc.). On the other hand, there are electronic elements such as power electronics, signal conditioning, transistors, etc. These electrical-electronic elements have the function to interface the actuators and sensors to the control interface/hardware elements.
- **Control interface/Computing hardware:** These elements are counters, timers, microprocessor, analogue to digital converter and vice versa, digital input/output, data acquisition, etc. Control interface hardware permits the communication between the sensors with the control computer and furthermore, the communication between the control computer and the actuator. Control algorithms are introduced by the control computing hardware, which uses the measurements made by sensors, to produce control actions that are going to be applied by the actuators.
- **Computer:** These elements are the hardware/software elements used to conduct computer-aided dynamic analysis, design, simulation and optimization, fast control of prototypes, control, etc.

2.2 Simulation

In order to reduce product development time and cost, companies are moving from traditional prototyping and testing towards a simulation-driven design process. This process reduces the need for expensive and time-consuming physical prototypes, allowing engineers to predict product performance with computer models (Thermal analysis 2003). Figure 2 shows the evolution of design process.

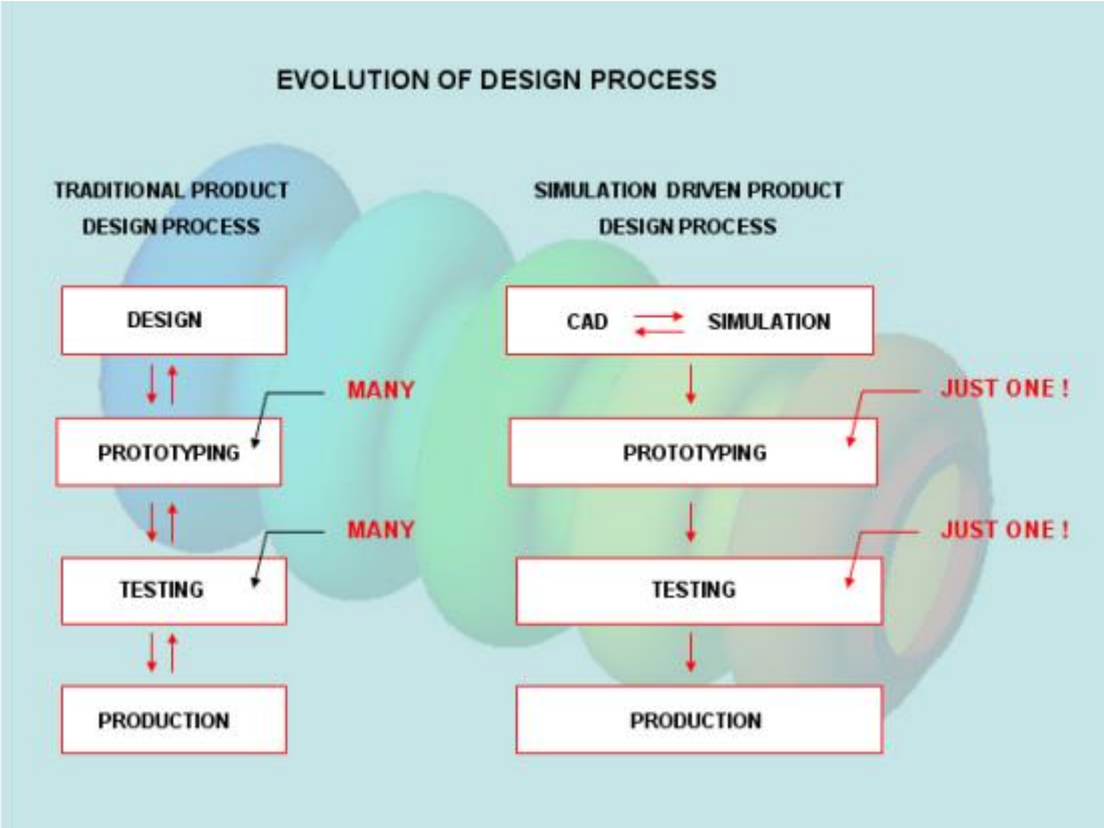

Figure 2: Evolution of design process (retrieved from Thermal analysis 2003)

In the mechatronic devices serval analyses can be conducted, which can be classified in the mechanical, electrical and control fields.

2.2.1 Finite element simulations (Mechanical)

As stated by Pelz "a particularly graphic form of structural modelling is to break down mechanical structures into finite elements for the modelling of continuum mechanics called meshing, and both geometric dimensions and topological information are important" (Pelz 2003).

Each element has its own matrix based on its material parameters and geometry, while the connections between the elements and the system matrix are derived from the topology. Sometimes manual checking of the meshing has to be done, to ensure for instance that the elements have a correct shape and size. Table 1 shows the input and output data of these simulations. Finite element simulations are used for a huge variety of mechanical simulations, and the most significant ones are listed below:

- <u>Static simulation</u>: Its main characteristic is that the applied load does not vary based on time, ignoring the inertial and damping forces. There are two types of static simulations: linear and non-linear. In the linear static simulations the relationship of the load applied to an object and the response of the object (displacement, stress, etc.) is linear (this is a good approach if only the elastic zone of the material is analysed). However, if the analysed part suffers plastic deformation, the relationship is not linear (it is given by the stress/strain curve of the material).

- Dynamic simulation: This kind of simulation is used when the applied forces are not constant over time (for examples the bumping forces on a vehicle running on a bumpy road). This means that these simulations are more complex than static simulations, since the time is added.
- Thermal simulation: Thermal analysis calculates the temperature and heat transfer within and between components in a machine/component and its environment. This can be a steady state or a transient analysis. Steady state is based on the assumption that enough time has passed for heat flow to stabilize (so the temperatures remain constant). On the other hand, an analysis of heat flow changing with time is called transient thermal analysis, as for example, the analysis of a coffee pot kept hot by a heating plate.
- Frequency analysis: The main objective of this analysis is to get the natural frequencies of a structure. This is important since if one of the excitation frequencies equals the natural frequency of the system, resonance will occur (which may cause premature failure of the machine/element).
- Buckling simulation: When a slender component is subjected to compression, buckling can occur before plastic compressive deformation. To achieve which are the buckling modes of a structure and its critical force, a buckling analysis must be carried out.
- Fatigue simulation: Fatigue is the weakening of a material caused by repeatedly applied loads (even if these are much smaller than the yield strength). It is the progressive and localized structural damage that occurs when a material is subjected to cyclic loading. Using fatigue simulation tools, the damage percentage on a component can be calculated after a certain number of oscillating cyclical forces.
- Fluid dynamics simulation: This type of simulation uses numerical analysis and data structures to solve and analyse problems that involve fluid flows. Computers are used to perform the calculations required to simulate the interaction of liquids and gases with surfaces defined by boundary conditions.

- Electromagnetic simulation: Electromagnetism is widely used in many electronic devices (e.g. electric motors and generators, relays, solenoids…). To calculate forces accurately and visualize complex magnetic fields, finite element magnetic simulations are used. These simulations are widely used by electronic device manufacturers.

Table 1: Input and output data in finite element simulations

Input data	Output data
• Geometry • Mesh • Material´s properties depending on the simulation type (Young´s modulus, conductivity, SN curve…) • Boundary conditions • External excitation or load depending on the simulation type (Force, heat source, magnetic field…)	• Dependant on the simulation type (stress, displacement, temperature distribution, natural frequency, buckling mode…)

2.2.2 Multibody simulation (Mechanical)

Multibody simulation is a method of numerical simulation where multibody systems are composed of various rigid or elastic bodies. Connections between the bodies can be modelled with kinematic constraints (such as joints) or force elements (such as spring dampers). Furthermore, frictional contacts can sometimes be modelled by unilateral constraints and Coulomb-friction. Multibody simulation is a useful tool for conducting motion analysis, and to evaluate characteristics of comfort, safety, and performance (Larsson 2001). Table 2 shows the input and output data of these simulations.

Table 2: Input and output data in multibody simulations

Input data	Output data
• Geometry/kinematic chain • Actuators/external loads • Physical properties of the bodies (mass, inertia, friction coefficient…)	• Kinematic sequence of the system

2.2.3 Electronic circuit simulation (Electronic)

Electronic circuit simulation uses mathematical models to replicate the behaviour of an actual electronic device or circuit. Simulating a circuit’s behaviour before actually building it can

greatly improve design efficiency by making faulty designs known as such, and providing insight into the behaviour of electronics circuit designs.” Typical features of electronics simulators include schematic editors and on-screen waveform display, which allow engineers to efficiently modify a circuit´s properties and to check the effects on the output. Normally generic electrical components are stored in the software´s libraries. These libraries typically include generic components such as transistors, resistors, capacitors, inductors and transformers, as well as current and voltage sources (Alimeling & Hammer 1999).
If an actuator is used in the electrical scheme, physical dynamic simulations can often be carried out combined with the electrical simulations. Table 3 shows the input and output data of these simulations.

Table 3: Input and output data in electronic circuit simulations

Input data	Output data
• Electronic devices and their properties • Connections	• Voltage • Current • Circuit´s behaviour

2.2.4 Control logic simulations (Control)

In order to properly understand the operation of a programmable logic controller (PLC), it is necessary to spend considerable time programming, testing, and debugging PLC programs. PLC systems are inherently expensive, and down-time is often very costly. In addition, if a PLC is programmed incorrectly it can result in lost productivity and dangerous conditions.” PLC simulation software allows engineers to write, edit and debug programs written using a tag-based format. With this kind of simulations, programmers can try all the possible scenarios in the virtual world efficiently, by changing logic instructions and programs and then running again the simulation to check how those changes affect the operation (Dougall, 1997).
On the other hand, these simulations can be used to carry out virtual commissioning of PLCs. As stated by Hoffmann et. al. virtual commissioning is “the process of testing manufacturing systems and associated control programs through simulation before the real system is realised” (Hoffmann 2010). Table 4 shows the input and output data of these simulations.

Table 4: Input and output data in control logic simulations

Input data	Output data
• Control logic program	• Behaviour of the controlled system

2.3 Simulation software

Simulation software is based on the process of modelling a real phenomenon with a set of mathematical formulas. It is, essentially, a program that allows the user to observe an operation through simulation without actually performing that operation (Pritsker & Alan 1979).

For simulating mechatronic devices several software packages can be used and some of the most relevant ones are explained in the following lines:

2.3.1 Simulia Abaqus

Routine and sophisticated engineering problems can be completed using Abaqus FEA software, which offers powerful and complete solutions. Abaqus offer four different solver technologies, based in finite elements (Simulia 2011):

- **Abaqus/CAE:** It offers the option to create, edit, monitor, diagnose and visualize advanced Abaqus analysis.
- **Abaqus/Standard**: This software is used for low speed dynamic and static cases where stress solutions with high accuracy are really important.
- **Abaqus/Explicit**: It is used for high speed and non-linear analysis.
- **Abaqus/Multiphysics**: It offers the capability to solve multiphysics problems.

2.3.2 Matlab/Simulink

Simulink offer the possibility to understand and analysed complex systems by simulating block diagrams. Complex systems can be modelled in one tool, using industry proving components from multiple domains including control systems, communication systems, mechanical systems and electrical systems. By just pressing the play button, it can be simulated the used model. Errors in the system can be found, by going back and forward through the simulation to understand the behaviour of the system. Simulink solver allows the simulation of an extensive range of analogue, digital, mixed signal and multi-frequency systems (MathWorks 2004).

2.3.3 Siemens Mechatronic Concept Designer

Mechatronics Concept Designer allows working with mechatronic products. Siemens Mechatronic Concept Designer offers the possibility to work in the domains of mechanics, electronics and control. The software enables 3D modelling and simulation of concepts with multi-body physics and automation-related behaviour typically found in mechatronics products.

With Mechatronics Concept Designer there is the possibility to work in parallel with the mentioned three domains design disciplines. The concept design is focused on mechanical components, actuators, sensors and motion.

2.3.4 Modelica

As stated by Fritzson "Modelica is an object-oriented, declarative, multi-domain modelling language for component-oriented modelling of complex systems, e.g., systems containing mechanical, electrical, electronic, hydraulic, thermal, control, electric power or process-oriented subcomponents" (Fritzson 2012).

It is generally used for simulations but it has also the possibility to optimize models.

Modelica offers the possibility to model in the domains of mechanics, electronics and control, visual causal hierarchical component modelling, typed declarative equation-based textual language and hybrid modelling and simulation.

2.4 Virtual confidence

Oscarsson et. al. stated that "the results of the simulation must predict the real results to be useful. If it is that way, it is said that virtual confidence is achieved, which means that the results are reliable. Use of virtual manufacturing can be classified in different levels with respect to the confidence you have in the results from simulation, and to which extent you can rely on CAE technology. The term virtual confidence (VC) captures this level of trust and utilization of virtual engineering" (Oscarsson 2016).

- Level 0: when CAE technology is not available.
- Level 1: when CAE technology is available but due to its immaturity not used in industry for the development processes.
- Level 2: when CAE technology is complementary to physical testing and support decisions.

- Level 3: when CAE technology is used for product-, process- and resource development. Results from CAE are used as a base for a majority of decisions. Still in this level, physical prototypes can be used as a complement to virtual verification.
- Level 4: when CAE technology is commonly used, and project gates are closed based on results from CAE. Serial (hard) tools etc. are ordered based on simulation results. There is no need of physical prototypes for tests or verification.

According to Oscarsson et. al. there are two main requirements to achieve virtual confidence. The first requirement is the possibility to link data from real experiments to virtual models, because the reality confirms the models. The second requirement is to be able to identify, link and classify products, processes and resources that are referenced by models. Products consist of parts, processes have sub-processes, and resources such as machines have various functions and capabilities, all interacting with each other (Oscarsson 2016).

2.5 Model verification & validation

As mentioned in the section 2.4, a good way of making product development more efficient is to achieve the level 4 of virtual confidence. To do so, the ability of the virtual model to represent in a correct way the behaviour of the real system is vital. In this field the concepts of model validation and verification turn out to be of major importance.

According to Hillston “by its nature a model is more abstract than the system it represents. Viewed in one way, abstraction, and assumptions we make to achieve it, eliminate unnecessary detail and allow us to focus on the elements within the system which are important from a performance point of view (which also decreases computational time); viewed in another way, this abstraction process introduces inaccuracy. Some degree of inaccuracy may be necessary, desirable even, to make the model solution tractable and/or efficient. Inevitably some assumptions must be made about the system in order to construct the model. However, having made such assumptions it must be expected to put some effort into answering questions about the goodness of our model” (Hillston 2003).

As described by Hillston (2003), the goodness can be judged in two steps: First, it has to be checked that the model implements the assumptions correctly (model verification), and second, that the assumptions which have been made are reasonable with respect to the real system (model validation).

2.5.1 Model verification methods

"Model verification is intended to ensure that the model does what it is intended to do" (Hillston 2003).

- Continuity testing: At an abstract level a model can be thought to be a function that relates input values to output values, and in most cases it is expected that function must be continuous. In other words, it is not expected that a slight change in an input value will result in very large changes in the corresponding output value. For any parameter, a slight change in input should generally produce only a slight change in the output. Having huge changes in output might be an indication of a possible error.
- Degeneracy testing: Degeneracy testing checks that the model works for the extreme values of input parameters. Even though extreme cases may not represent typical cases, degeneracy testing can help the modeller to find errors that would not otherwise have been discovered.
- Consistency testing: This testing assumes that for most models and systems, similarly loaded systems will exhibit similar characteristics, even if the arrangement of the workload varies. Consistency tests are used to check that a model produces similar output data for different input parameter values that have similar effects. For example, in a static analysis the use of symmetry (half of the part and half of the load) should give a practically identical result as when the whole system is analysed. If the model output shows a significant difference, either it should be possible to explain the difference from more detailed knowledge of the system, or the possibility of a modelling error should be investigated.

2.5.2 Model validation methods

"Validation is the task of demonstrating that the model is a reasonable representation of the actual system: that it reproduces system behaviour with enough fidelity to satisfy analysis objectives. To do so, the aspects that need to be validated must be compared to a reliable source" (Hillston 2003).

Depending on the source to which the hypotheses and results are compared, there are three approaches to model validation. These approaches are:

- Expert intuition: Many times to contrast the simulation outcomes with the intuition of experts that have worked with similar systems over the years is an appropriate screening method to check if the results are logical or not. However, when it comes to the accuracy of the results, this approach is often not enough.

- Real system measurements: As stated by Hillston “comparison with a real system is the most reliable and preferred way to validate a simulation model. In practice, however, this is often infeasible either because the real system does not exist or because the measurements would be too expensive to carry out. Assumptions, input values, output values, workloads, configurations and system behaviour should all be compared with those observed in the real world” (Hillston 2003).

  Having access to the real system measurements also offers a great advantage: Several parameters of the input data and the model itself can be calibrated so that the initial model can be refined (which means that it reproduces the real system more accurately). A virtual model can be calibrated with respect to the real output parameters or the real input parameters. For example, when a static analysis of a part is carried out these two approaches can be followed: it can be an input data focused calibration (e.g. geometry of a virtual model is refined to accommodate the fabrication tolerances that might occur during the fabrication process, the friction coefficient is calibrated with the real measured data, etc.) or on the contrary an output data focused calibration (e.g. the stresses and the displacements are measured on the real system, and the non-fixed parameters of the model (such us simplifications, finite element mesh…) are iterated until the output data matches the real measurements).

- Theoretical results/analysis: Sometimes simplicity of the system allows a more abstract representation to provide a crude validation of the model. In particular, if the results of a theoretical analysis based on the scientific laws coincide with model output it may be taken as evidence that the model behaves correctly.

In the table 5 a general review of the different validation methods can be found:

Table 5: Model validation methods

Model validation methods		
Expert intuition	Real system measurements	Theoretical results/analysis
The most basic validation. Useful for detecting early stage big errors. Definitely not enough for a good virtual confidence.	Essential for a good level of virtual confidence. It offers a reliable source at the expense of a higher cost.	If the studied system is simple this approach might be enough to have a fairly good virtual confidence. However, it is not useful for complex systems.

3. Approach

In this chapter the mechatronic device and the models used for simulating it are explained. Furthermore the simulation setups are also clarified.

3.1 Working method

In order to achieve the main goal successfully, at first a working method with several steps or phases was defined. These phases constitute a logical order to follow and can be found in the following lines:

1. Definition of the working cycle: The only data to define the working cycle of the real machine was a video provided at the beginning of the project. Based on the information offered on the video, the cycle has been self-defined by realistic assumptions.
2. Creation of the necessary components: As it can be seen in the section 3.3, the model offered by Volvo was not created to do simulations in Mechatronic Concept Designer, so offered many limitations. To cope with this problem, new more detailed components were self-created in SolidWorks. Also here, it was tried to create as faithful models as possible to the original.
3. Isolated simulations of the sub-systems: Once the sub-system components were created and imported to Siemens Mechatronic Concept Designer, the main two functional sub-systems of the machine were simulated on their own. Dynamic analyses were done to calculate the dynamic performance of the system.
4. Simulation of the whole system: The next step was to make a simulation of the whole system, with the two sub-systems working together. The main goal was to calculate the time of the working cycle, based on the dynamic performance calculated in the previous section.
5. Conclusions & virtual commissioning discussion: Once all the relevant simulations were conducted it was time to accomplish the main goal of the project, which is to analyse the virtual confidence. Here the advantages and disadvantages (limitations) will be discussed and some improvement lines will be set.

3.2 Mechatronic Concept Designer simulations

First of all, it is important to clarify the simulation type that can be carried out by Mechatronic Concept Designer (MCD). This software is used to simulate the behaviour of mechatronic systems by carrying out dynamic multibody analyses; the physical properties of the bodies (mass, inertia, material…), the joints, actuator properties and operations will be the input data, while the simulated movement sequence will be the output (with special regard to the kinematics and operation times). The virtual confidence of this movement sequence will be analysed, which means how close to reality it can be.

Mechatronic concept designer offers a range of different analysis features, divided in the mechanical, electrical and control domains. First of all, the physical properties of the system can be defined by the "rigid body" and "collision body" features:

- Rigid body: A rigid body is an element of the system that can move around and act as mass in the mechatronics system. Once a geometric object is defined to be a rigid body it responds to forces, e.g. it is falling down due to the gravity.
- Collision body: A collision between multiple bodies can be simulated efficiently with this feature. The bodies can be assigned different collision shapes, depending on the complexity that is wanted to achieve (the most complex shape being the mesh). Furthermore, this feature offers the option to set the material properties of the collision, such as static/dynamic friction and restitution (bouncing).

On the other hand, the kinematic behaviour of the system is defined by a wide range of connectors that MCD offers:

- Hinge joint: The hinge joint performs a fixed rotation along an axis
- Sliding joint: The sliding joint performs a fixed linear movement along a vector but does not allow the bodies to rotate with respect to each other.
- Ball joint: A ball joint constraints two bodies connected at a point, both can rotate freely.
- Cylindrical joint: The cylindrical joint rotates like the hinge joint but can also telescope along the rotation axis.
- Fixed joint: The fixed joint provides a rigid connection of two rigid bodies. It can also be used to fix a component on the global coordinate system.
- Angular spring joint: The angular spring joint generates a torque when angles between bodies change.

- Linear spring joint: The linear spring joint generates an opposing force when bodies travel apart. This constraint works similar like the angular spring joint. While the angular spring joint is looking at the angle of two rotating objects, the linear spring joint is looking at the distance between two moving objects.
- Angular limit joint: The angular limit joint prevents bodies from rotating past a given angle.
- Linear limit joint: The linear limit joint prevents bodies from moving beyond a given distance. For linear joints, the distance is determined by offset points – the points specified in the global space. When the constrained objects move, the assigned point maintains its relative position to the object.
- Transport surface: The transport surface is a physical property that turns any selected flat surface into a kind of "conveyer belt". Once another body lies on this surface it is transported into a specific direction.
- Cam: A cam forces two axes to maintain a relationship determined by a graph (motion profile).
- Gear: A gear forces two rotating axes to maintain a constant turn ratio. It is a special case of a cam defined by a motion profile with constant speed.

The electrical domain of MCD consists basically of sensors and actuators.

- Collision sensor: A collision sensor is attached to a body and detects the presence of other items within its boundary. With custom defined shapes, different types of sensors even ultrasound or inductive sensors can be replicated. However, they are ON/OFF sensors, which means that the only output they provide is if they are activated or not.
- Speed control actuators: The speed control sets the movement of an axis (which is identified by a joint) to a predefined constant speed. Depending on the properties of the real actuator limits for the acceleration, jerk and torque can be applied.
- Position control actuators: The position control moves an axis to a predefined position with a predefined speed. This feature is intended to simulate servo-motors. As in the previous case, limits for the acceleration, jerk and torque can be applied.

Finally, the control domain includes the feature "operation", used to create logic sequences:

- Operation: This feature receives information from the collision sensors and commands the actuators. Mechatronics Concept Designer distinguishes between three kinds of operations:

- For a purely time based operation both the start and end is determined by the time.
- A time based operation that is linked to a position constraint with defined speed and position has an undetermined duration. Time is automatically derived based on the runtime situation.
- An event based operation is invoked by a certain event, e.g. a collision with a collision sensor. In this case both the point of time for start and end are not predetermined. The earliest point of time the trigger condition is considered is the start time on the time scale.

3.3 Original machine & model

The machine that has been simulated in this project is a turn-table used in a sheet metal forming station at the AB Volvo plant in Umeå. The plant is shown in the figure 3 and the turn-table, which was used as reference object in this study, is highlighted in the red square. The individual model of the turn-table is shown in the figure 4.

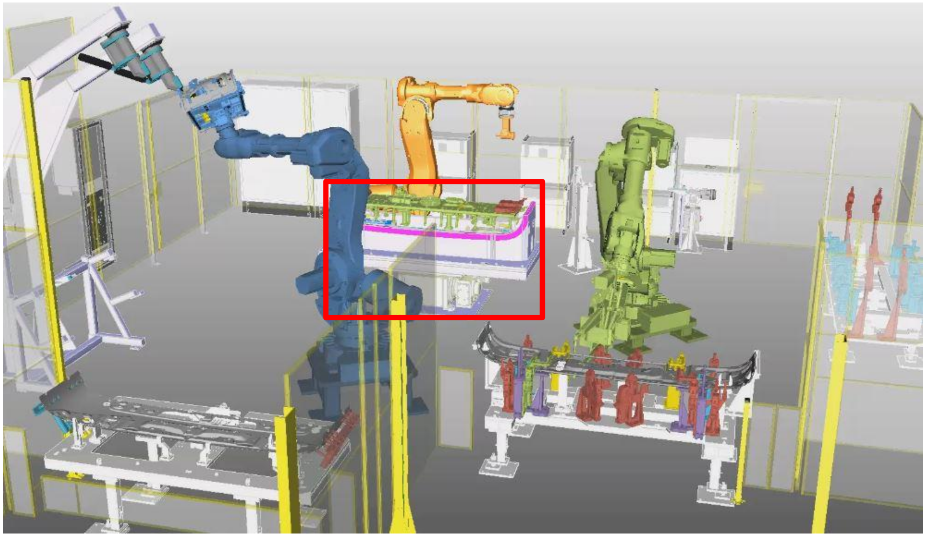

Figure 3: The plant from Volvo (in Umeå), where the orange robot is working on the turntable

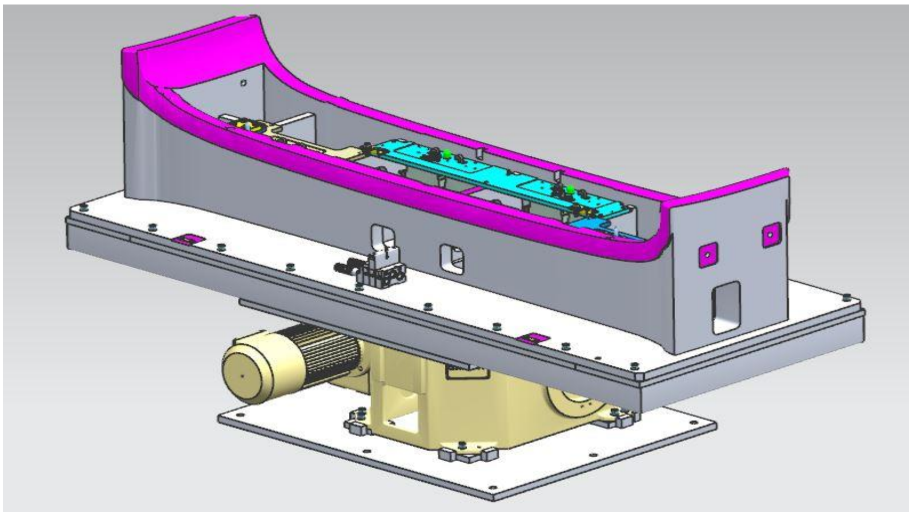

Figure 4: The turntable that has been simulated

3.3.1 Sub-systems

Two main sub-systems of the turn-table are the clamping mechanism (which will clamp the working part to the table) and the turning mechanism (which will turn the table 180 degrees to let the orange robot work on the other side of the part). These sub-systems are shown in figures 5 and 6:

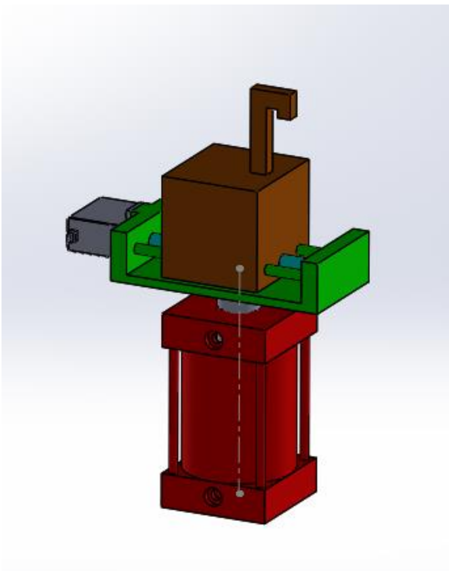

Figure 5: The clamping mechanism

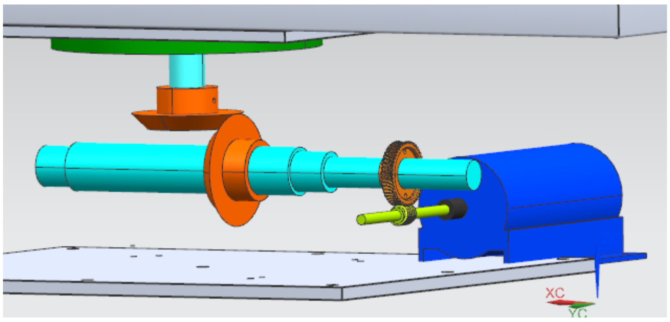

Figure 6: The turning mechanism

3.3.2 Working cycle

The working cycle of the robot is defined below. As said in the section 3.1 this cycle has been self-created making realistic assumptions based on the plant simulation video. Note that the colour of the robots can be seen in the figure 3.

1. The green robot places the working part on the table.
2. The sensors on the top of the table that check if the part is correctly positioned are activated.
3. The clamping mechanism is activated, which clamps the part to the table.
4. The green robot retracts from the turn-table.
5. The orange robot works on the sheet metal part, turn-table waits.
6. The turning mechanism is activated, turn-table rotates 180º.
7. Robotic arm works on the sheet metal part, turn-table waits.
8. The turning mechanism is activated, turn-table rotates 180º.
9. Robotic arm works on the sheet metal part, turn-table waits.
10. The turning mechanism is activated, turn-table rotates 180º.
11. Robotic arm works on the sheet metal part, turn-table waits.
12. The turning mechanism is activated, turn-table rotates 180º.
13. The green robot grasps the working part.
14. The clamping mechanism is activated, which releases the part.

3.3.3 Limitations

As stated in the section 3.1 the original model that was provided was not created to do component level simulation, but station level simulations. In the station level simulations the specific functioning of the internal components of the machine is not relevant, because the performance specifications of the machine are used as input data for a higher level simulation (station level simulation). After a brief inspection of the original model, some significant limitations were found that made the component level simulation on this model practically unviable. These are listed below:

- The functional sub-systems of the machine were not completely defined or too simplified; the clamping system consisting on pneumatic cylinders and ball spindles was modelled as an unique solid part (figure 7), which made impossible its simulation. On the other hand, the kinematic chain of the turning system was not even defined: just the horizontal axle was featured, and the vertical axis and the motor´s gearbox were missing (figure 8).

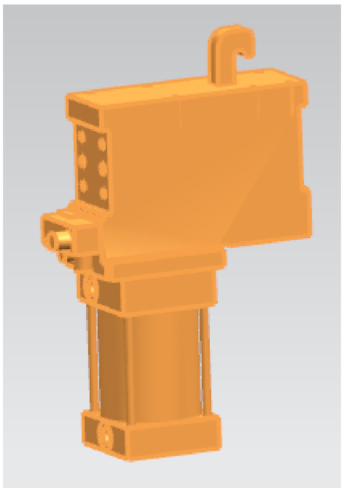

Figure 7: Clamping mechanism as an unique solid part

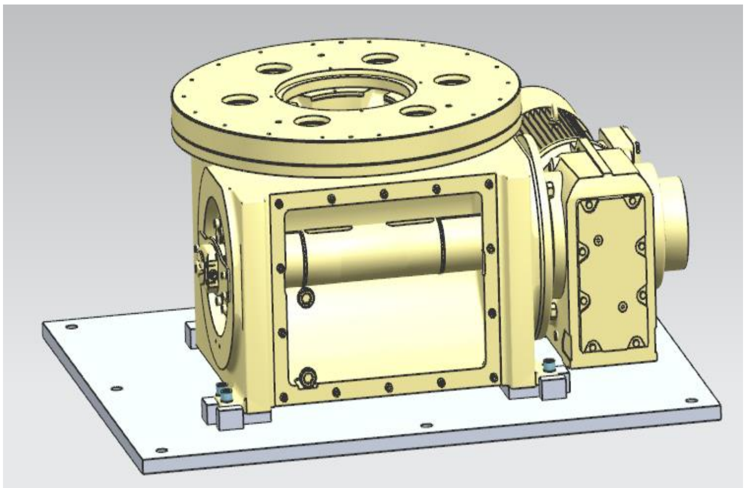

Figure 8: Turning mechanism with just the horizontal axle

- Some geometric data of the components was missing. This means that the components just represented 3D shapes with no axes, centre-points, mid-points, etc. These features are necessary to create the mechanical joints used to define the mechanical behaviour of the system.
- The mass of the components was missing; all the parts had by default 1 kg of mass (this could be changed and user defined, but the material and inertia data was missing too).

All this limitations forced us to create our own self-created detailed sub-systems, which are explained in the next section.

3.4 Detailed models of the sub-systems

The limitations offered by the original model made compulsory the creation of more detailed sub-systems of the machine. As stated before these sub-systems are the clamping system and the turning system. The models have been created by SolidWorks and after imported to Mechatronic Concept Designer. This process has some requirements to import all the necessary data to the latest software correctly. First of all, when creating the part in Solidworks just creating the geometry is not enough, since Mechatronic Concept Designer itself does not offer the material assignment to parts. That is why the material should be assigned in the CAD software (in this case Solidworks). On the other hand, when it comes to the exported file formats, these can be exported both in the SLDPRT (SolidWorks part) or in the STEP formats. When these files are opened in Mechatronic Concept Designer automatically a copy of them will be created in the JT format. This format will be used to add components to the assembly. Finally, the assembly will be saved as a UG part file (it can also be saved as a JT file, but it gives problems when opening the components so it should be avoided). Table 6 shows the importing process.

Table 6: Format used in the software

CAD software (SolidWorks)	⟶	Simulation software (MCD)	
Components	Components	Components	Assembly
SLDPRT	SLDPRT, STEP	JT	UG

3.4.1 Clamping system

The clamping system consists of four clamps located on the top of the turn-table and sensors. As it can be seen in the figure 9 there are 4 inductive sensors located in the clamping mechanism. These are used to check if the sheet metal part is located accurately, preventing manufacturing errors. The upper white component is just a vague approximation of the manufactured part. It is featured in the simulation to make the sensors work and to check the performance of the clamps.

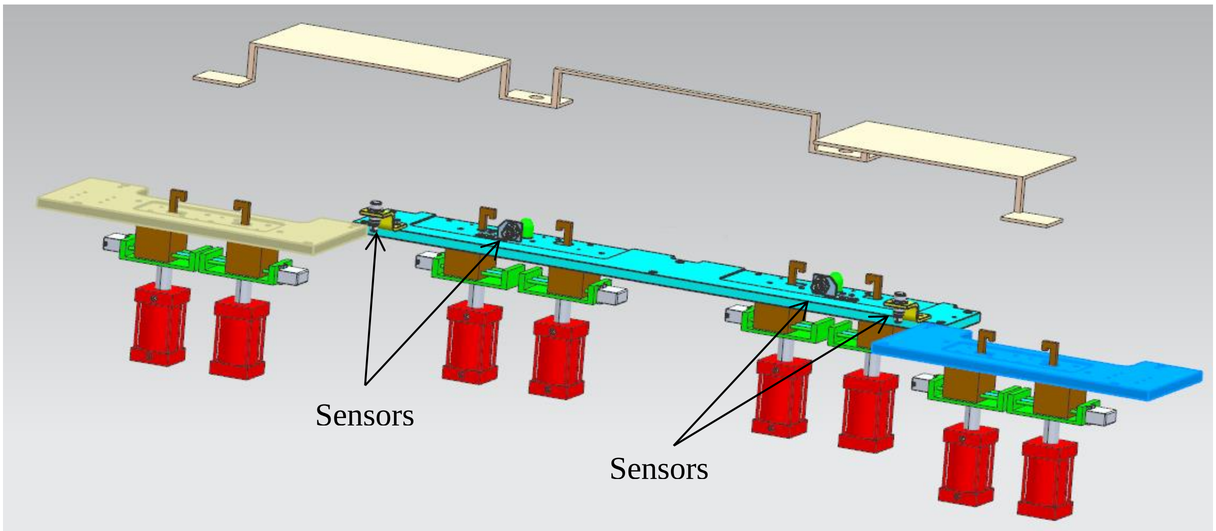

Figure 9: The clamping system and the sheet metal part (above) before clamping it to the system

Each clamp is made of a pneumatic cylinder (powered by a pump) and a ball spindle (powered by a servo-motor). It has two movement axes; vertical (cylinder) and horizontal (ball spindle). The specifications of these actuators have been assumed based on their size and similarly used actuators, and are shown in the tables 7 and 8.

Table 7: Specifications of the pneumatic cylinders

Pneumatic cylinder	
Chamber area (Forward/backward)	1962/1472 mm^2
Maximum pressure	0.5 MPa
Pump´s flow rate	31500 mm^3/s
Maximum force (Forward/backward)	981/736 N
Speed (Forward/backward)	16.05/21.40 mm/s

Table 8: Specifications of the ball spindle

Ball spindle	
Motor´s maximum speed	1500 rpm
Motor´s nominal torque	0.02 N
Motor´s inertia	12 kg·mm^2
Thread pitch	1 mm

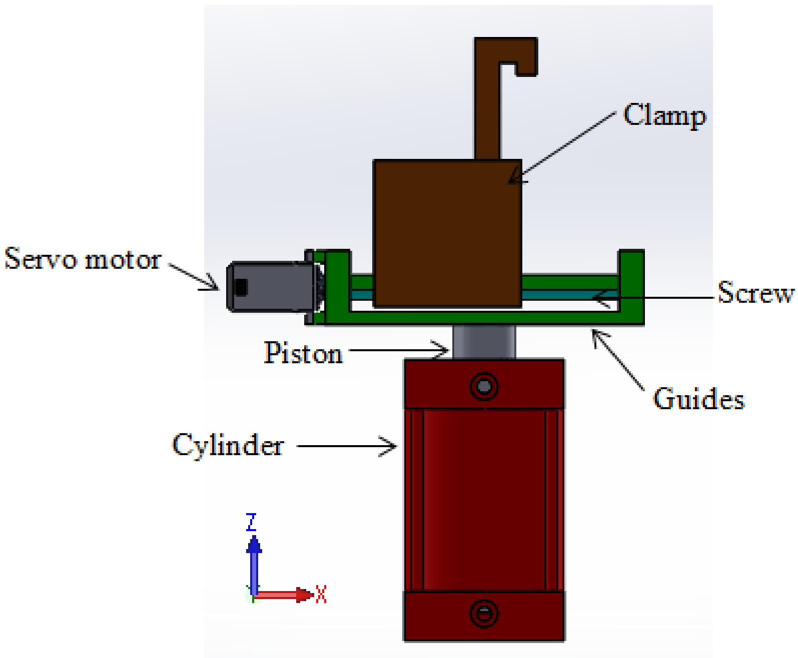

Figure 10: All the parts of the clamping mechanism

The working cycle of one clamp consists of the following consecutive sequences:

1. The green robot places the working part on the table
2. The sensors on the top of the table that check if the part is correctly positioned are activated.
3. Clamp goes 12 mm forward in the x axis

4. Clamp goes 16 mm down in the z axis, part is clamped
5. Wait
6. Clamp goes 16 mm up in the z axis
7. Clamp goes 12 mm backward in the x axis

3.4.2 Turning system

The turning system is composed by several parts. To complete this system, a transmission formed by gears has been created. A servo-motor is used to give motion to the turning system. A reductive box is connected to the motor to reduce the velocity and increase the force. This reductive box is formed by a worn drive (worm screw with worm wheel) where the worm screw is attached to the motor using a coupling. All the parts of the turning system can be seen in figure 11:

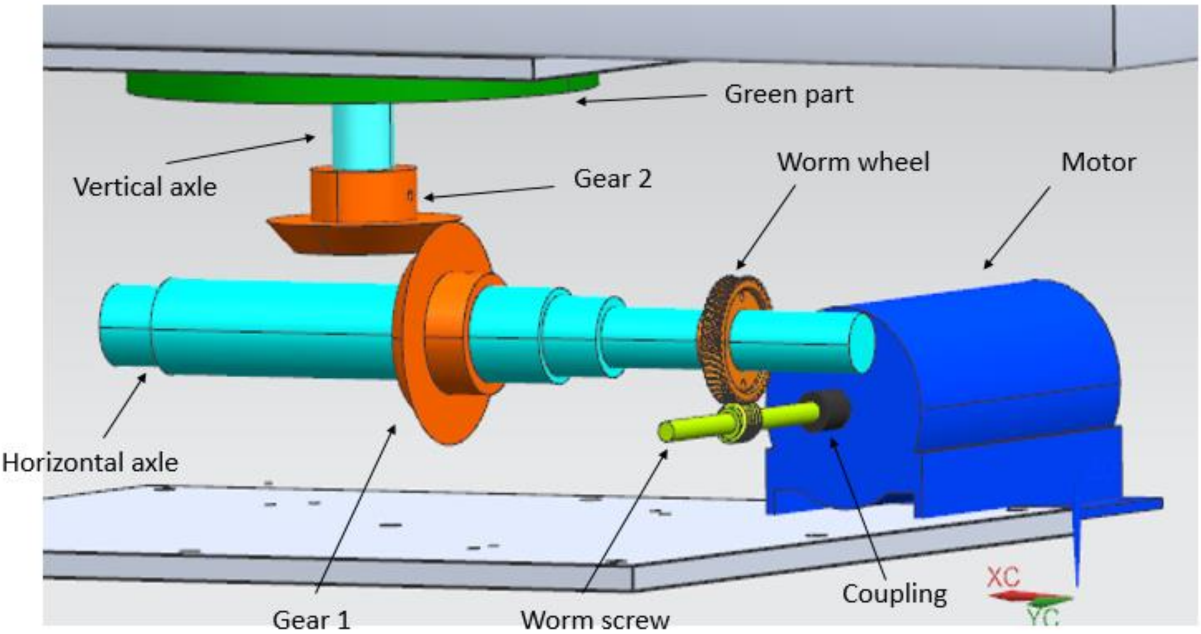

Figure 11: All the parts of the turning system

The motor that has been used is a servo-motor which has the characteristics shown in table 9:

Table 9: Specifications of the motor

Motor	
Motor´s maximum speed	1500 rpm
Motor´s nominal torque	25 Nm
Inertia	0,075 kg·m^2

The worm drive passes the motion to the horizontal axle where gear 1 is attached. Then, gear 1 makes contact with gear 2, which is attached to the vertical axle, transmitting the motion to this second axle. The type of gears that have been used are conical. The vertical axle shown in figure 12 contains six cylindrical arms that are used to attach to the green part shown in figure 13. This second part is attached to the principal table, which makes the table move (rotation in the z axle).

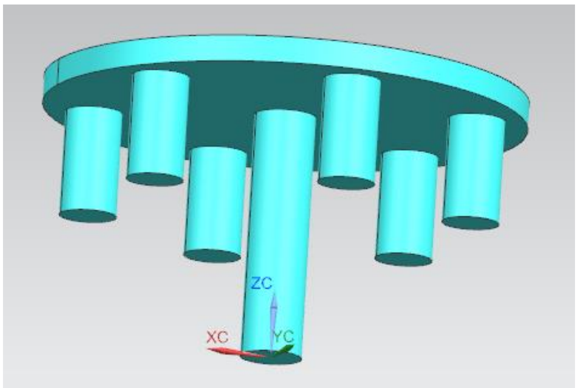

Figure 12: The vertical axle with its six cylindrical arms

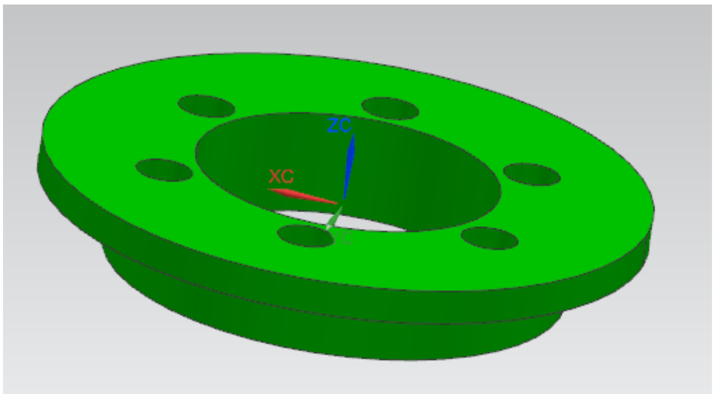

Figure 13: The green part

As it can be seen in the figures 12 and 13, the cylindrical arms of the vertical axle match with the holes of the other part. So when the vertical axle rotates, the other part would also rotate. Then, as the principal table is attached to the green part, it starts rotating also along with the mentioned two parts. This rotation is the function of the turning system. First, the table rotates 180 degrees. Then, it stays on that position until the robot completes its operations. After that, the table turns again to the initial position to wait until the robot completes its operations.

The green part is the one which makes contact with the box that covers the transmission system, so friction is created there. The boxes that have been used to cover the transmission (the big box) and the worm drive (the small box) can be seen in figure 14 along with the whole table:

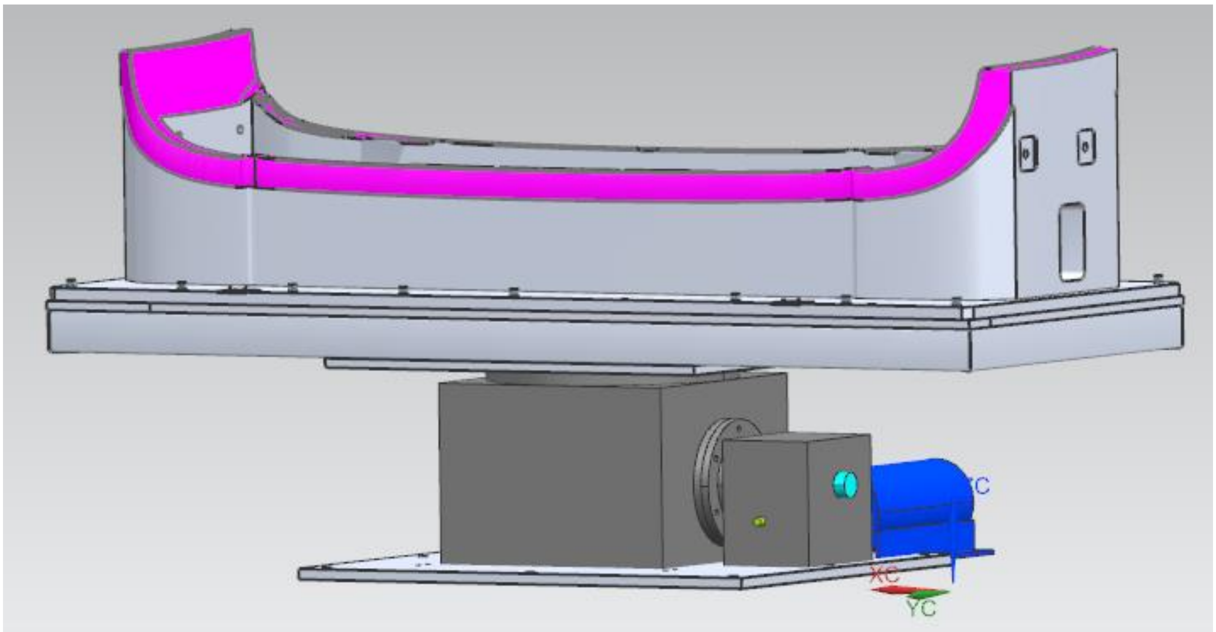

Figure 14: The whole table with the mentioned boxes used to cover

If the outside boxes that are used to cover the transmission taken away, the table would look like shown in figure 15:

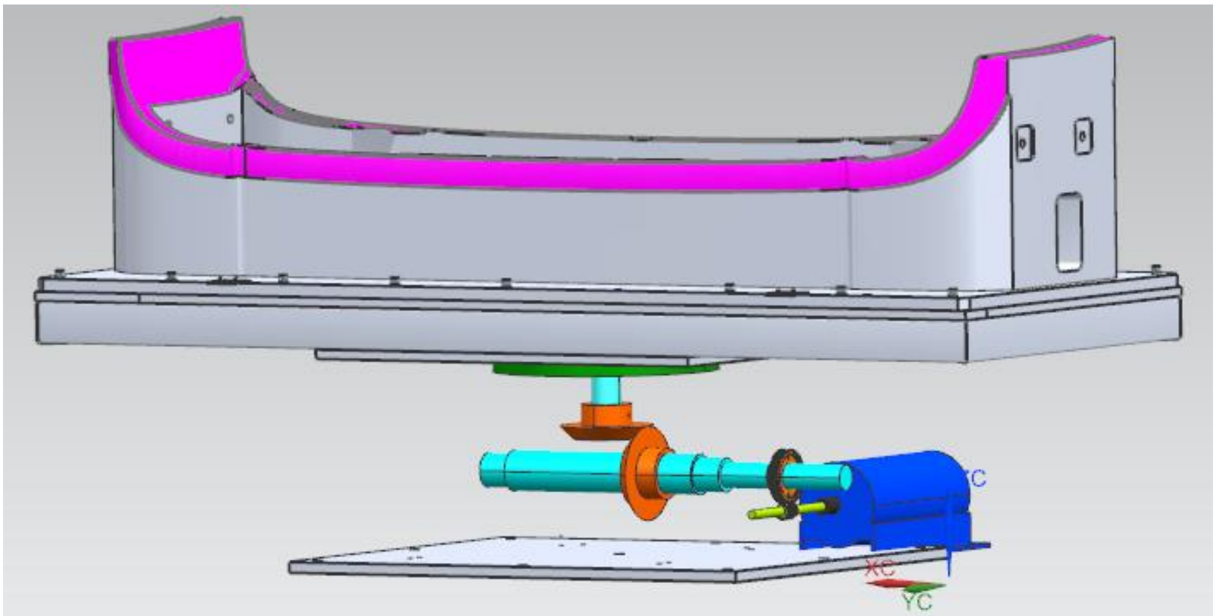

Figure 15: The turntable without the boxes

3.5 Simulations set ups

3.5.1 Clamping system

- Mechanical:

First of all, all the parts have been defined as rigid bodies, and they have been assigned with automatic mass and inertias by the geometries and materials defined in SolidWorks. On the other hand, collision bodies have been defined: this feature has been applied to the support bases of the table, the contact surfaces of the work part and the contact edge of the clamps. These are the surfaces that will be pressed together during the clamping.

On the other hand, the kinematic chain has been defined by the following joints:

1. The cylinders have been fixed, as if they were fixed to a table.
2. The piston is joined with the cylinder with a sliding joint.
3. The piston and the part containing the horizontal guides are fixed together
4. The guide part and the screw are linked with a hinge joint
5. The upper clamp slides on the guides with a sliding joint
6. The upper clamp and the screw are linked with a cam link. This link enables the relation of the angular movement of the screw and the linear movement of the clamp. As the thread pitch is 1 mm, the movement relation is 0.00278 mm/degree.

- Electrical:

The clamping system has two different actuators. One is the pneumatic pump that moves the cylinders, which will be controlled by a relief valve. The other one is the servo-motor of the ball spindle, controlled by an internal encoder. Both have been defined by position control actuators. To simulate the properties of the actuators their speed and torque have been limited. On the other hand, the system has two sensors on each side of the table. In reality they are inductive sensors, which means they do not collide physically with the detected object. However, MCD only provides collision sensors. To solve this problem, the detection line of the sensor has been custom made, to make it longer and simulate an inductive sensor.

- Control:

The clamping cycle consists of four operations, two for the clamping and two for the releasing. First the clamp goes 12 mm in the x direction; to do so, the motor and the screw rotate 4320º (this operation will start when the inductive sensors are triggered). After that, the piston goes down until it squeezes the part. When all the turning operations are finished, the releasing starts: first the piston goes up again to its initial position. After that the clamp goes back 12 mm in the x direction (- 4320º for the motor). This operations are shown in the figure 16.

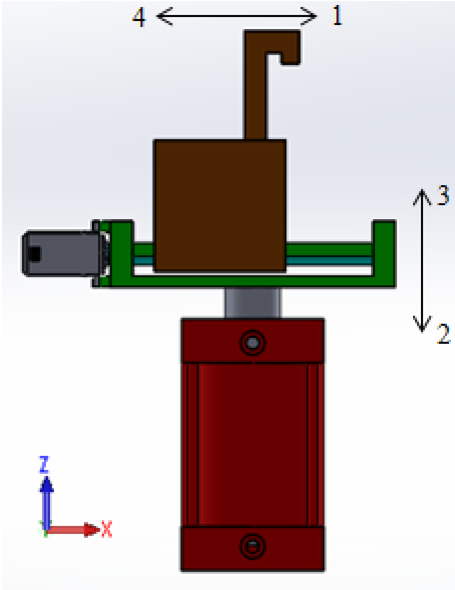

Figure 16: The clamping mechanism´s movements

3.5.2 Turning system

- Mechanical:

To complete the simulation of the turning table, first, the basic physics have been defined. Here, the different parts have been defined as rigid bodies. All the parts have been defined as rigid body individually except in two cases. The two cases are these ones:

1. All the parts of the table (not the parts of the transmission), have been defined as one rigid body.
2. The green part and the vertical axle have been defined as one rigid body because their motion is the same (like if they were fixed).

Then, some joints and constraints have been defined. The constraints that have been used are the hinge joint and the fixed joint. Three hinge joint constraints have been applied. These have been the parts which have been defined with the hinge joint (considering that the first named part is the attachment and the second one the base):

1. The worm screw with the small box that covers the worm drive.
2. The horizontal axle with the big box that covers the transmission.
3. The green part which is attached to the table with the big box that covers the transmission.

On the other hand, seven fixed joints have been defined. These have been the parts which have been defined with the fixed joint (considering that the first named part is the attachment and the second one the base):

1. The base of the whole table (the rectangular part below the table) has been fixed.

2. The small box that covers the worm drive with the base.
3. The worm wheel with the horizontal axle.
4. The gear 1 with the horizontal axle.
5. The gear 2 with the vertical axle.
6. The table with the green part and vertical axle (as they are one rigid body).
7. The big box that covers the transmission with the base.

Finally, two couplers (gear relations) have been defined. The first one is between the worm screw and the worm wheel. The worm wheel has 48 teeth while the worm screw has one start worm, so that, the relation defined has a ratio of 48:1. The value 48 has been given to the axle of the worm screw (master axle) and the value 1 has been given to the axle of the horizontal axle (slave axle) because the worm screw would rotate 48 times with one rotation of the worm wheel.

The second gear relation has been defined between the conical gears. Both gears have the same sizes except the middle hole which is used to introduce the gears in the horizontal and vertical axles. These holes, otherwise, don´t affect to the transmission relation so as the sizes of the gears are the same, the ratio used in the relation has been 1:1. The master gear or axle is of the horizontal axle, and the slave, of the vertical axle.

- Electrical:

In the case of the turning system, a servo-motor has been put as the actuator. This actuator gives to the worm screw the required motion of the system. The velocity provided by the motor is 1500 rpm.

A position control actuator has been used to complete the turning of the table. As it has been said before, the actuator is a servo-motor which gives motion to the worm screw. So that, the actuator has been stablished in the axle of the worm screw. The initial conditions that have been defined in the position control actuator have been that the initial velocity and position are 0.

- Control:

The turning system works by turning the table 180˚ and then by turning again to the initial position in the opposite direction. When the table locks the part that is going to be manufactured, the turning system waits until the robot finishes working on the part.

The operations that have been applied are based on the mentioned position control actuator. As it has been explained before the actuator is located on the worm screw and this screw would need 48 turns, in order to produce a turning in the worm wheel. To do so, the conditions that have been applied in the operation 1 have been that the velocity is 1500 rpm and the screw must rotate 8640º, because the table needs to rotate 180˚ and the worm screw needs 48 turns to complete a turning of the vertical axle, so 180·48 = 8640º.

The other operation (operation 2) that has been used has the same conditions, except the position, where the destination is 0º.

Once the operations have been defined, the sequences of the turning system have been made. Firstly, when the robot finishes working on the part, operation 1 is applied and the table turns 180˚. Secondly, the robot works on the part again and after that, operation 2 is applied while the table returns to the initial position.

The same operations (1 and 2) with same characteristics are repeated one second time until the work on the part is finished. Finally, another robot takes the part out. The cycle starts again when another part is put on the table.

3.5.3 Whole turn-table

Finally, the setup of the whole turn-table has been done. This is basically the sum of the two sub-system setups, but coordinated together. First the clamping operation happens, then the turning sequence follows and finally the releasing operation finishes the cycle.

3.6 PLC program implementation in MCD

The control features have been directly entered using the operation feature to create a time based cycle. However, to increase the virtual confidence by closing the gap between reality and simulations real PLC programs can be used to control operations in Mechatronic Concept Designer. Due to unavailability of compatible programming tools this option has not been developed in this project. However, using external resources the requirements to implement a real PLC program in Mechatronic Concept Designer have been evaluated.

The first step would be to create the corresponded PLC program of the working cycle (in reality) in a software as an OPC (Open Platform Communications) server. Then, the variables which are in the PLC program would be configured in the OPC and from the OPC would be sent to MCD.

Once the PLC program and the variables have been sent to MCD, it must be opened the option “OPC client parameters” which can be found inside “external controller”. After selecting this, a window is opened where the user select the OPC sent previously. The parameters used in the OPC must all be selected.

The next step is to open the option “OPC signal mapping” which can also be found inside “external controller”. Here, the window shown in figure 17 would be opened.

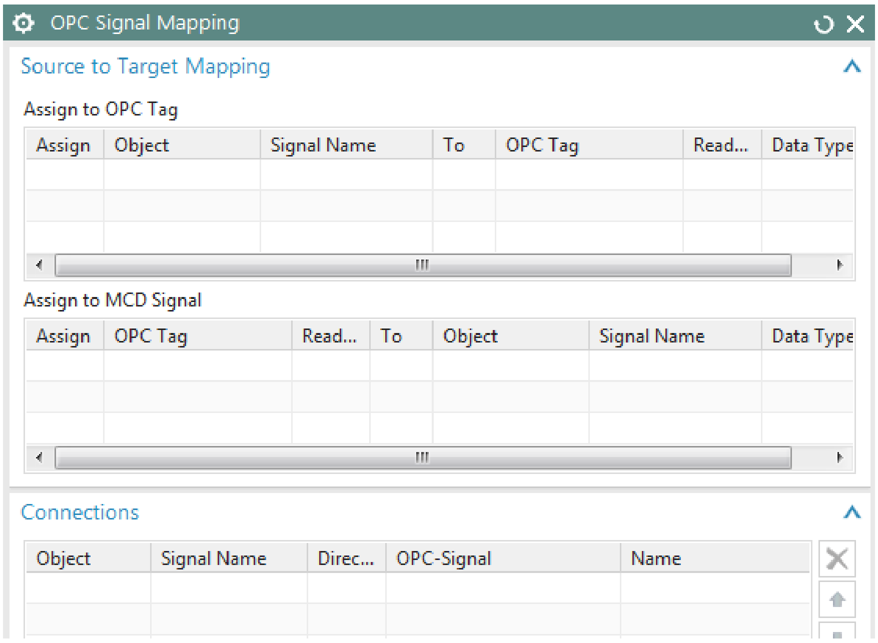

Figure 17: Window of the “OPC signal mapping”

In the window, there are two different “sources to target mapping”. The first one is “assign to OPC tag” where the parameters that would have been sent previously can be found. So, a parameter would be selected and after that, in the second source called “assign to MCD signal”, the chosen parameter would be put in the gap “OPC tag”. Finally, in this second source, the gap “object” would be filled in by putting the name of the part that would require the selected parameter. These operations must be repeated with all the parameters, by assigning them correctly to the corresponded parts. All the assigned operations would be shown in the gap called connections.

To conclude, data would be sent to the PLC created at the beginning, from the OPC to MCD. After completing this, the system is ready for the simulations.

4. Results

4.1 Limitations to implement friction joints

One interesting feature of the software Mechatronic Concept Designer is that it can simulate the friction between two collision bodies (bodies which collide together). To include friction properties of e.g. ball spindles, worm gears, conical gears, etc. was a good way to increase the virtual confidence. However, many limitations have been found when it comes to implement this feature in the mentioned components, which suggests that this software is able to use friction in a much more simple set of conditions.

4.1.1 Hinge joints/cylindrical sliding joints

When trying to simulate friction with the collision body feature in hinge joints or cylindrical sliding joints many problems occurred. The axis can be modelled with cylindrical shape. The hole, however, does not work with cylindrical shapes; instead of creating an empty cavity with the hole´s edges, it creates a solid collision body into the hole, which the axis cannot penetrate. The closest shape that can merely model the hole is the mesh, but this also presents big drawbacks that make the use of friction in these joints practically impossible. Figure 18 shows the meshed hole with the smallest element size.

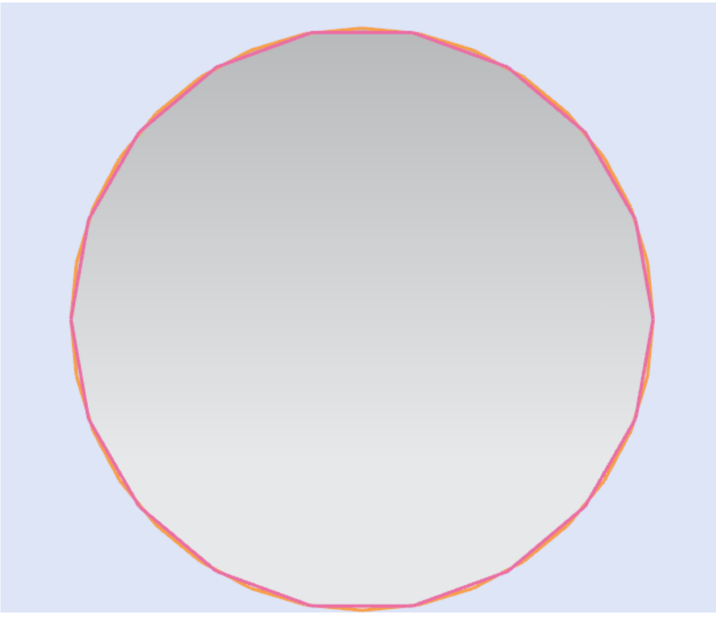

Figure 18: The meshed hole with the smallest element size

The hole is meshed as a high order polygon, whose edges interfere with the axis and make the simulation unstable and the results useless. Figure 19 shows the simple hinge joint used for this analysis. Figures 20 and 21, show the results of speed and torque (which are unstable and useless). As a matter of fact, changing the friction coefficients from 0.7 (used for the analysis, typical for steel-steel contacts) to any other value does not alter its results, which is completely illogical. This supports the statement that friction is not supported in these joints.

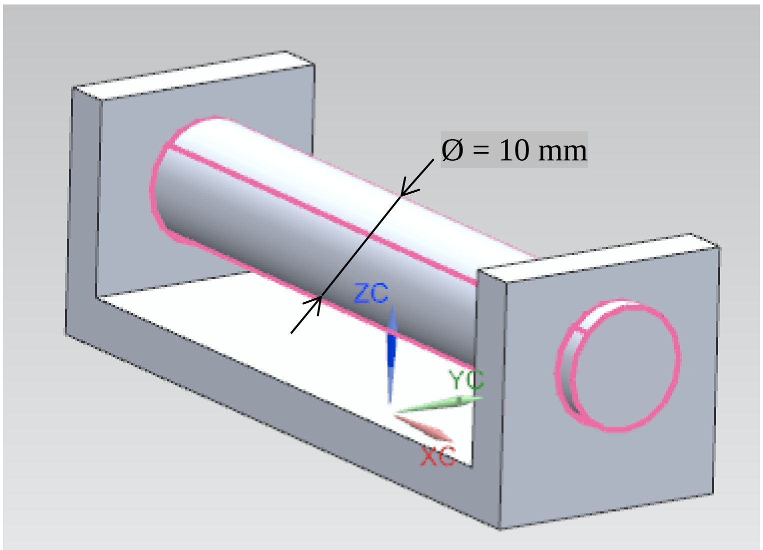

Figure 19: The used hinge joint for the analysis

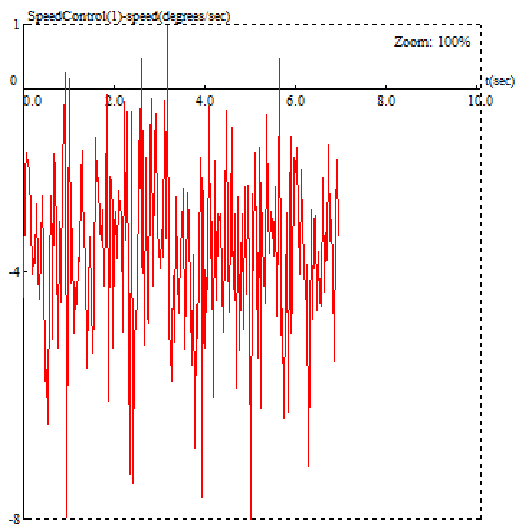

Figure 20: Results of the speed obtained with friction

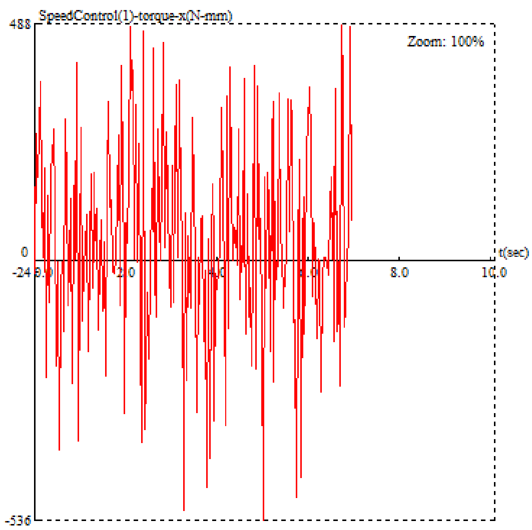

Figure 21: Results of the torque obtained with friction

4.1.2 Gears

A dynamic analysis has been conducted with gears to see how the friction could affect in the acceleration and the duration of the operations. The gears or the model that has been used to complete the simulation can be seen in figure 22.

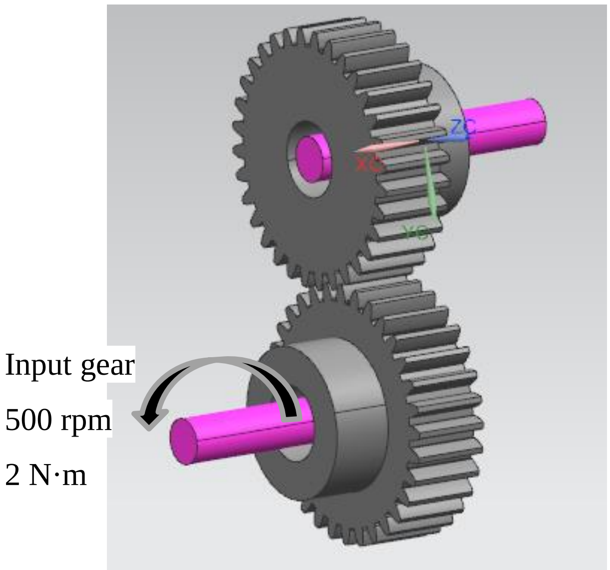

Figure 22: Gears used to complete the dynamic analysis

The two axles have been constrained as fixed joints. Then, two identical gears have been put as hinge joints in the axles. The relation between the gears has been completed using collision body.

A speed control has been applied to the input gear where they have been defined a speed of 500 rpm and a torque of 2 N·m, as shown in the figure 22. Finally, using collision material, friction has been defined with a value of 0.7, which is a typical value for steel-steel contacts (The Engineering ToolBox 2018). The result of the speed/time graph of the input gear is shown in figure 23:

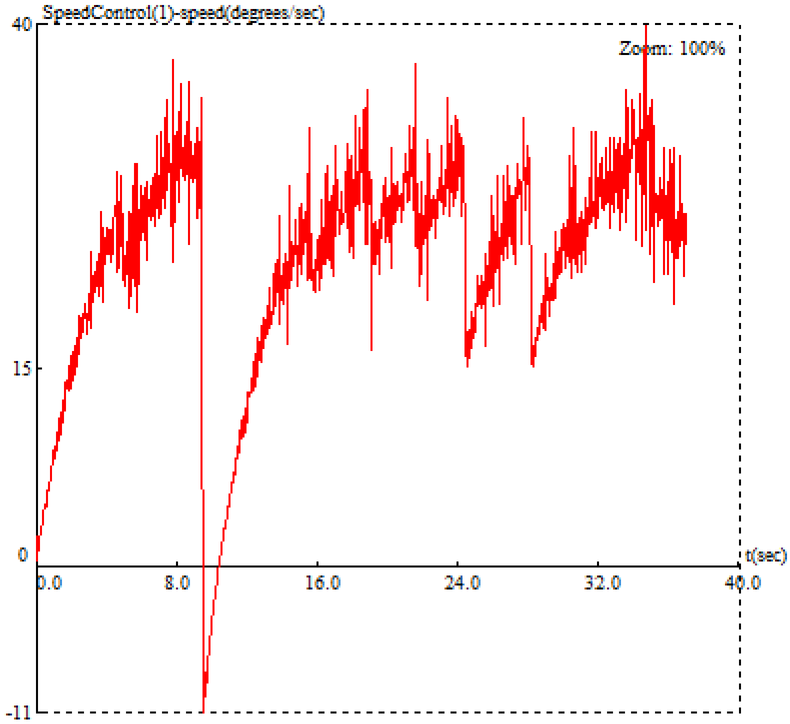

Figure 23: Results of the speed/time graph with friction between gears

It can be seen that the results obtained have no sense as the speed goes up and down instead of increasing at the beginning until it achieves the required speed. After seeing that simulations don´t work properly with friction, friction hasn´t been used in the dynamic simulations.

4.1.3 Rolling resistance

Rolling resistance, sometimes called rolling friction or rolling drag, is the force resisting the motion when a body (such as a ball, tire, or wheel) rolls on a surface. It is mainly caused by non-elastic effects; that is, not all the energy needed for deformation (or movement) of the wheel, roadbed, etc. is recovered when the pressure is removed. Rolling resistance is not supported by MCD, because it only supports rigid bodies, and no deformable bodies at all.

4.2 Dynamic analysis

Mechatronic Concept Designer does not introduce any torque limit in the control operations; this means that when simulating the working cycle, the kinematic variables (position, speed and acceleration) must be entered manually (and not calculated by the torque of the actuator and inertia of the system). However, this calculation can be done independently, by using a speed control actuator with a torque limitation (as said before this limitation is not available in the operation settings). This analysis just calculates the time that it takes to accelerate the motor to its maximum speed, by setting a torque limit and inertias of the system. This analysis has been carried out with the two electric motors of the system.

In order to analyse the accuracy of the dynamic analysis, the same calculus has been done analytically, as a power transmission exercise. The comparison between simulated and analytical results is presented in section 5.1.

4.2.1 Main turning motor

In the case of the servo-motor used to give motion to the turning system, a dynamic analysis has been done to see how the system acceleration is and how much time the motor needs to achieve a speed of 1500 rpm (used for the simulations).

First, a speed control has been put in the axle of the servo-motor. The characteristics or inputs of the servo-motor used in the speed control have been the inertia, speed and the limit torque. The inertia that has been used is 0.076 $kg \cdot m^2$, while the speed and torque used have a value of 1500 rpm and 25 $N \cdot m$.

The next and final step has been to use the runtime inspector by analysing and obtaining the speed graph of the speed control. The graph that has been obtained can be seen in figure 24.

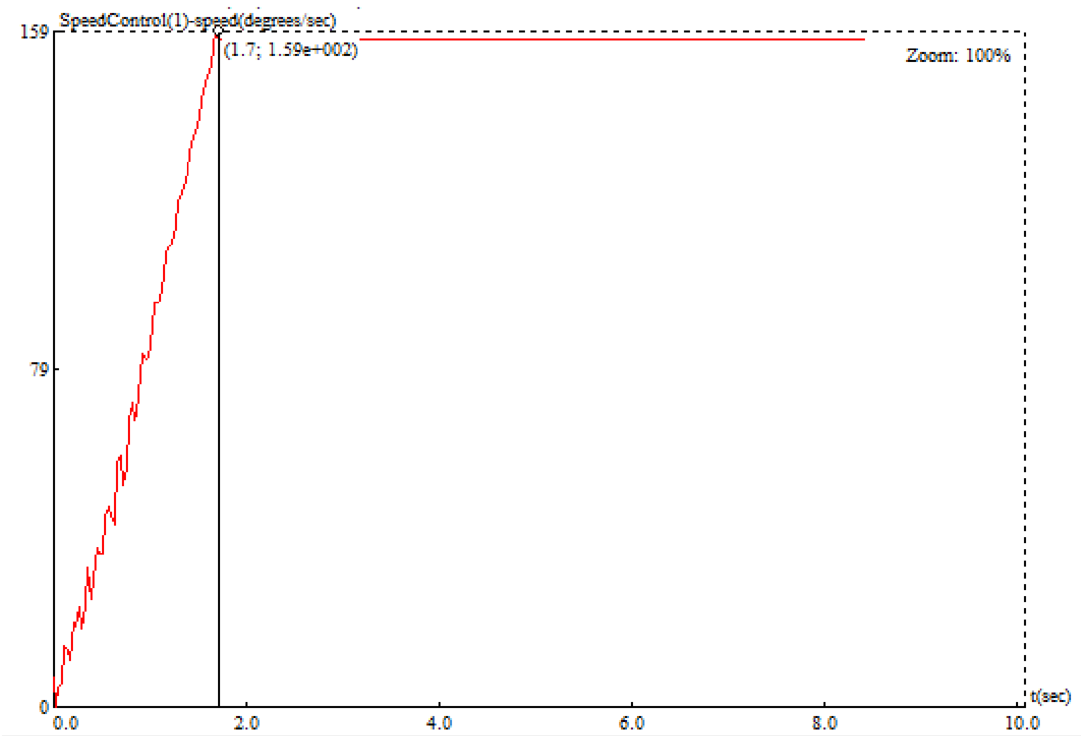

Figure 24: The speed graph obtained of the speed control

The units of the speed in the graph are in rad/sec although the graph says that there are in degrees/sec. The graph shows that the turning system needs 1.7 seconds to accelerate the system until a speed of 1500 rpm is achieved. As it has been said before, the speed in the graph is of 159 rad/sec but converting it to rpm, the value of the speed is 1500 rpm.

Finally, the angular acceleration has been obtained using the speed and time:

$$\alpha = \frac{w}{t} = \frac{159}{1,7} = 93.5 \text{ rad/s}^2 \tag{1}$$

Analytical results

First of all, a scheme with the components of the turning system has been done to see more clearly how the system is composed with different angular velocities, moments and inertias. Figure 25 shows the scheme of the system:

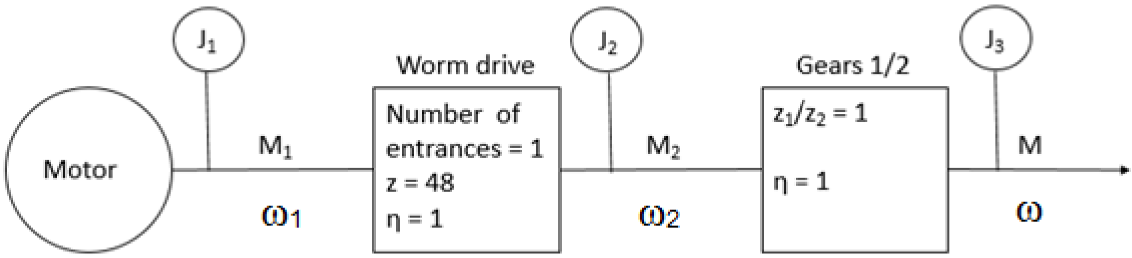

Figure 25: The scheme of the turning system

After completing the scheme, the equivalent inertia of the system has been calculated from the equation 2:

$$\frac{1}{2} \cdot J_{\mathrm{eq}} \cdot \omega_1^2 = \frac{1}{2} \cdot J_1 \cdot \omega_1^2 + \frac{1}{2} \cdot J_2 \cdot \omega_2^2 + \frac{1}{2} \cdot J_3 \cdot \omega^2 \tag{2}$$

The inertias J_1, J_2 and J_3 mentioned in the equation 1 are formed by:

- J_1: The inertias of the motor (J_M) + worm screw (J_S)
- J_2: The inertias of the worm wheel (J_W) + horizontal axle (J_H) + gear 1 (J_{G1})
- J_3: The inertias of the gear 2 (J_{G2}) + vertical axle (J_V) + table (J_T)

Using the mentioned inertias, equation 2 is obtained, which has been used to calculate J_{eq}:

$$\frac{1}{2} \cdot J_{\mathrm{eq}} \cdot \omega_1^2 = \frac{1}{2} \cdot (J_{\mathrm{M}} + J_{\mathrm{S}}) \cdot \omega_1^2 + \frac{1}{2} \cdot (J_{\mathrm{W}} + J_{\mathrm{H}} + J_{\mathrm{G1}}) \cdot \omega_2^2 + \frac{1}{2} \cdot (J_{\mathrm{G2}} + J_{\mathrm{V}} + J_{\mathrm{T}}) \cdot \omega^2 \tag{3}$$

$$J_{\mathrm{eq}} = 0.238 \text{ kg} \cdot \text{m}^2$$

Once the equivalent inertia has been obtained, the angular acceleration has been obtained from the equation 4, using the moment given by the motor (25 N·m) and the equivalent inertia (0,238 kg·m^2):

$$M = J_{\mathrm{eq}} \cdot \alpha \tag{4}$$

$$\alpha = 105 \text{ rad/s}^2$$

4.2.2 Ball-spindle motor

The ball spindle motor has a maximum speed of 1500 rpm and an inertia of 12 kg·mm^2. On the other hand, automatically assigned inertias and masses have been used for the system´s components. The maximum torque of the motor has been limited to 0.02 N·m (nominal torque). As it will be discussed further in the section 4.3 using the nominal torque as the torque limit, is a conservative assumption. Figure 26 shows the speed/time graph that is obtained in this system.

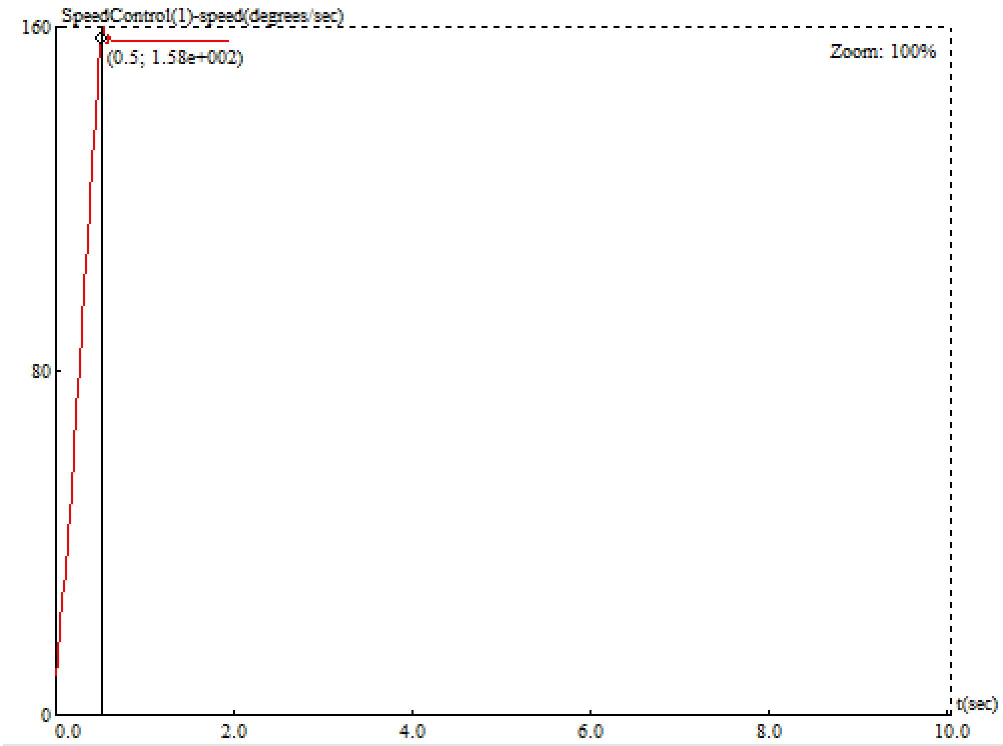

Figure 26: The obtained speed/time graph of this system

It can be seen that it takes 0.5 s to accelerate the system to its maximum speed (1500 rpm). This gives an angular acceleration of 314.16 rad/s^2.

Analytical results

The kinematic chain has been modelled manually as shown in the image 27.

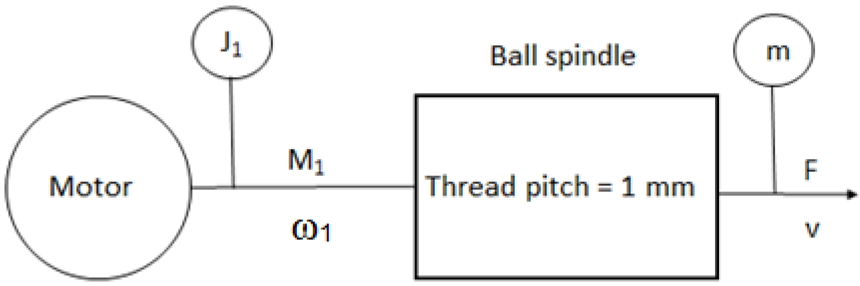

Figure 27: Kinematic chain of the ball spindle

- ω_1=1500 rpm
- $$v = \frac{p(\text{mm}) \cdot \omega_1(\text{rad/s})}{2\pi \cdot 1000} = 0.025 \text{ m/s} \tag{5}$$

To calculate the acceleration manually, the equivalent inertia of the system has been calculated, applying the principle of energies.

$$\frac{1}{2} \cdot J_{\text{eq}} \cdot \omega_1^2 = \frac{1}{2} \cdot J_1 \cdot \omega_1^2 + \frac{1}{2} \cdot m \cdot v^2 \quad (6)$$

The inertias and masses J_1 and m mentioned in the equation 6 are formed by:

- J_1: The inertias of the motor + screw (60.96 kg mm^2)
- m: Mass of the clamp (4.54 kg)

$$J_{\text{eq}} = 6.142 \cdot 10^{-5}\ \text{kg} \cdot \text{m}^2$$

Once the equivalent inertia has been obtained, the angular acceleration has been obtained from the equation 7 using the moment given by the motor (0.02 Nm) and the equivalent inertia.

$$M = J_{\text{eq}} \cdot \alpha \quad (7)$$

$$\alpha = 325.63\ \text{rad/s}^2$$

4.3 Working cycle analysis

As said before, while simulating the entire cycle of the machine by sequence operations, actuators´ intrinsic properties (torque, power) cannot be entered; on the contrary, the kinematic variables of the system will be entered. To make this analysis as real as possible, accelerations will be limited to the values obtained in the dynamic analysis. The working cycle analysis is useful to help programmers foresee the problems of the PLC programs and check their correctness. The simulated cycle of the machine is shown below (notice that this is the same as the one defined in the section 3.3.2, but this time the simulated times have been added)

1. The green robot places the working part on the table (0 s)
2. The sensors on the top of the table that check if the part is correctly positioned are activated. (0 s)
3. The clamping mechanism is activated, which clamps the part to the table (1.73 s)
4. Robotic arm works on the sheet metal part, turn-table waits (2 s)
5. The turning mechanism is activated, turn-table rotates 180º (2.54 s)
6. Robotic arm works on the sheet metal part, turn-table waits (2 s)
7. The turning mechanism is activated, turn-table rotates 180º (2.54 s)
8. Robotic arm works on the sheet metal part, turn-table waits (2 s)
9. The turning mechanism is activated, turn-table rotates 180º (2.54 s)
10. Robotic arm works on the sheet metal part, turn-table waits (2 s)
11. The turning mechanism is activated, turn-table rotates 180º (2.54 s)

12. The clamping mechanism is activated, which releases the part (1.98 s)

The total simulated time of the work cycle is 21.87 s, considering that the waiting time of the turn-table is 2 s (this value has been randomly inputted). Not considering the waits, the total cycle time is 13.87 s.

5. Discussion

In the discussion, a thorough interpretation of the work is presented by evaluating the assumptions, the results and potential sources of error.

5.1 Virtual confidence discussion

The objective of this project was to analyse the virtual confidence that can be achieved with the turn-table model and the software Mechatronic Concept Designer. Each of them will be discussed independently in the following lines.

5.1.1 Model of the turn-table

As stated in the section 2.4 there are two main requirements to achieve a high level of virtual confidence with a simulation model. The first requirement for trusted models is the ability to link data from real experiments and operation to virtual models. The second requirement is to be able to identify, link and classify products, processes and resources that are referenced by models. In the case of the model that it was provided, none of these requirements was met. First of all, the only real data provided was a video of the real turn-table rotating, from which it was seen that in reality it takes 3.5 s to rotate. Simulated rotation time is 2.54 s, which differs 27% from reality, but the reason for this difference is unknown, due to the lack of specific real data (masses, inertias, torque, etc.); not having the physical data of the real components and the motor´s properties made a proper model validation impossible. On the other hand, the model did not have any of its provenience (such as the hypotheses considered while creating it) referenced. If we add the fact that the model was not really operative in the Mechatronic Concept Designer software, we get a very poor virtual confidence.

5.1.2 Mechatronic Concept Designer software

To analyse the virtual confidence that this software can provide, the limitations have been analysed in the mechanical, electrical and control areas.

Mechanical

First of all, when it comes to the ability to model kinematic chains, they can be modelled quite accurately since there is a wide variety of mechanical joints available (described in the section 3.2).

The main limitation from the mechanical point of view is that friction is featured in a very basic level in this software. The inability of using friction in hinge joints, sliding joints, screws, etc. makes the simulations to be less reliable. Actually, this drawback makes the simulated operations times shorter, because with no friction the system can accelerate faster.

When it comes to the accuracy of the dynamic analyses, these have proven to be quite accurate (compared to analytical results). However, the accelerations obtained in the simulation are slightly lower than the theoretical ones. This is because during the simulation the torque of the motor has momentary decreases (probably due to small instabilities of the simulation itself).

All in all, if the analysed system is a basic low friction one, the dynamic results obtained can be considered quite accurate. However, in almost all of the industrial mechatronic devices friction plays an important role. The inability to simulate it properly gives a poor virtual confidence to this software, from the mechanical point of view.

Electrical

The electrical domain of MCD consists of sensors and actuators. The “actuator” feature of the software offers a limited range of input data. Basically, when it comes to setting the intrinsic properties of the motors the only available options are to limit the maximum speed and torque. The option of defining the torque/speed curve of the motor and using it for the dynamic analysis would be a great improvement to improve the virtual confidence level. On the other hand, there is no way to simulate directly a pneumatic cylinder by setting its working pressure and flow rate. Instead, a linear actuator must be used, limiting its maximum force and speed.

On the other hand sensors can be defined using just the “collision sensor” feature. This can simulate properly binary object sensors (most commons in industry), as its only output will be triggered/not triggered. However, there is no possibility to simulate other kind of sensors, such as optic sensors that provide colour outputs, or ultrasound sensors that provide distance outputs.

Control

As stated in the section 4.2 when defining the control operations the torque cannot be limited. Instead, accelerations obtained in the dynamic analysis must be defined. When it comes to control simulation, Mechatronic Concept Designer offers a very high virtual confidence. First of all, the mechanical response of the system to the program can be seen in a very intuitive and graphic way, which can help programmers develop higher quality programs faster. On the other hand, a real virtual commissioning of a PLC program can be done, because this software accepts the real Siemens PLC programs that are used in the machines. This fact makes the virtual confidence increase a lot, as the same program that is simulated will be the one that control the real device.

5.2 Method discussion

The version of Mechatronic Concept Designer that has been used for this project is 10.0. However, at the time this report is written the latest version is 12.0 which might have improvements and more analysis features. Using the latest version the virtual confidence might be higher, but this is just a hypothesis.

On the other hand, when it comes to the model that has been used, the first one proved to be useless. This is why new models were created, and to do so all the properties of the system were assumed. These assumptions were made to be as realistic as possible, but they probably differ from the real system. If the accurate characteristics of the real system were known, much more realistic assumptions would be done, improving this way the virtual confidence.

The model validation has also been quite limited, as the data from real life performance was missing. The model has been “validated” by comparing it to analytical results, but this is definitely not enough to get a high virtual confidence.

The friction problems have not been analysed in the turn-table model. Instead, specifically created test models have been used. The reason for this was to simplify the analyses, obtaining more clear results. However, similar results were expected if the friction was tried in the turn-table model.

5.3 Technology, Society and the Environment

The outcome of this simulation related project is relevant for the whole community of engineering, as nowadays time and cost efficiency of product development processes is becoming a must. Specifically though, the results of this project might be very useful for future users of Mechatronic Concept Designer, as well as, for Siemens (they can use these results as a customer review, to improve their product).

From the environmental and economical sustainability point of view, improving simulation software can be really helpful. Reducing the need for physical prototypes makes the development process faster and cheaper.

5.4 Future work

The future work on the research about virtual confidence could focus on analysing other models and software. For the same kind of simulations, the following software is suggested:

- Simulink
- Simumatik 3D
- Modelica

It would be interesting also, if Siemens would include the following features in the future updates:

- Improved friction features for mechanical connections (gears, hinge joints, sliding joints, screws)
- Improved actuator feature, including the possibility to input the torque/speed curve
- Possibility to input motor´s properties in the operations.

6. Conclusions

In this chapter, the most important findings, key results and contributions of the project are summarised.

On the one hand, the original model provided for Volvo was intended to be used in another kind of simulators, and had huge limitations which made the multibody simulation impossible. The missing data from real experiments made the situation even worse, as a proper validation could not be carried out. That is why its virtual confidence has been considered to be very poor.

On the other hand, Mechatronic Concept Designer has shown its weaknesses and limitations on the mechanical and electric domains; the most important ones are the inability to implement friction properly and the inability to define the torque/speed curve of the actuators (only the maximum torque can be limited). Leaving all this problems aside, the dynamic calculus was quite accurate to the analytical results. However, the software really shines in its control domain, since giving the possibility to use real PLC programs can help programmers check their work faster and cheaper (virtual commissioning).

Finally, to get the whole picture about virtual confidence it is important to take into account that the multibody simulations carried out by Mechatronic Concept Designer are not enough to get a “total” virtual confidence (referring to level 4 of virtual confidence, for more information check section 2.4). Before manufacturing a machine many other aspects must be taken into account and simulated: for instance, the safety factors of the critical parts, thermal effects, natural modes (simulated by finite element simulators, such as Abaqus or Ansys) or the wear and life expectancy of the components (which can be approximated by KISSsoft). Once all these aspects have been simulated and validated with real experiments we will be able to get close to the highest level of virtual confidence.

References

Alimeling, J.H.,& Hammer, W.P. (1999), *PLECS-piece-wise linear electrical circuit simulation for Simulink*, retrieved from https://ieeexplore.ieee.org/abstract/document/794588/

Bishop, R. (2007), *Mechatronic Systems, Sensors, and Actuators.* Taylor & Francis Group

David J. Dougall (1997), *Applications and benefits of real-time IO simulation for PLC and PC control systems*, retrieved from http://www.sciencedirect.com/science/article/pii/S0019057897000335

Engineering.nyu.edu (2010), *Elements of Mechatronics*. [Online] Available at: http://engineering.nyu.edu/mechatronics/Description/elements.htm. Accessed 04/02/2018

Fritzson, P. (2012), *The Modelica Language and Technology for Model-Based Development* [online] Openmodelica.org. Available at: https://www.openmodelica.org/images/docs/Modelica-and-OpenModelica-overview-Peter-Fritzson-120328.pdf. Accessed 16/02/2018

Hillston, J. (2003), *Model Validation and Verification* [Online] Available at: http://www.inf.ed.ac.uk/teaching/courses/ms/notes/note14.pdf. Accessed 18/02/2018

Hoffmann, P., Schumann, R., Maksoud, T. M. & Premier, G. C. (2010), *Virtual Commissioning of Manufacturing Systems a Review and New Approaches for Simplification*. Proceedings 24th European Conference on Modelling and simulation. Retrieved from https://www.researchgate.net/profile/Andrzej_Bargiela/publication/235991499_Proceedings_of_the_24th_European_Conference_on_Modelling_and_Simulation/links/0deec516006eaa13e0000000/Proceedings-of-the-24th-European-Conference-on-Modelling-and-Simulation.pdf#page=194

JustScience. (2017), *Applications of mechatronics* [Online] Available at: http://www.justscience.in/articles/applications-of-mechatronics/2017/07/11. Accessed 05/02/2018

Larsson, T. (2001). *Multibody dynamic simulation in product development* (PhD dissertation). Luleå. Retrieved from http://urn.kb.se/resolve?urn=urn:nbn:se:ltu:diva-26621

MathWorks, 2004, *Simulink - Simulation and Model-Based Design*, [Online] Available at: http://users.isr.ist.utl.pt/~alex/micd0506/simulink.pdf. Accessed 21/02/2018

Morshedzadeh, I., Oscarsson, J., Ng, A. H. C., Jeusfeld, M. A. & Jenefeldt, A. (2016). *Real World Data Identification and Classification for Support of Virtual Confidence*. Presented at the 7th Swedish Production Symposium, Lund, Sweden, October 25-27, 2016. Retrieved from http://urn.kb.se/resolve?urn=urn:nbn:se:his:diva-13736

Oscarsson J., Jeusfeld M.A. & Jenefeldt A. (2016) *Towards Virtual Confidence - Extended Product Lifecycle Management*. In: Bouras A., Eynard B., Foufou S., Thoben KD. (eds) Product Lifecycle Management in the Era of Internet of Things. PLM 2015. IFIP Advances in Information and Communication Technology, vol 467. Springer, Cham

Pelz, G. (2003), *Mechatronic Systems - Modelling and Simulation with HDLs*, Wiley

Pritsker, A., & Alan, B. (1979), *Introduction to simulation and SLAM*, Wiley

Simulia, 2011, *Abaqus 6.11*, *Theory manual*, [Online] Available at: http://130.149.89.49:2080/v6.11/pdf_books/THEORY.pdf. Accessed 20/02/2018

SolidWorks Corp. (2010), *Thermal analysis* [Online] Available at: www.solidworks.com/sw/docs/thermal_2010_eng_final.pdf. Accessed 02/02/2018

The Engineering ToolBox, 2018, *Friction and friction coefficients*, [Online] Available at: https://www.engineeringtoolbox.com/friction-coefficients-d_778.html

Appendix 1 Work breakdown and time plan

In this chapter, the initial work breakdown and time plan for the project is compared with an updated version which shows how the project actually took place.

Figure 28 shows the initial Gantt chart, developed in January.

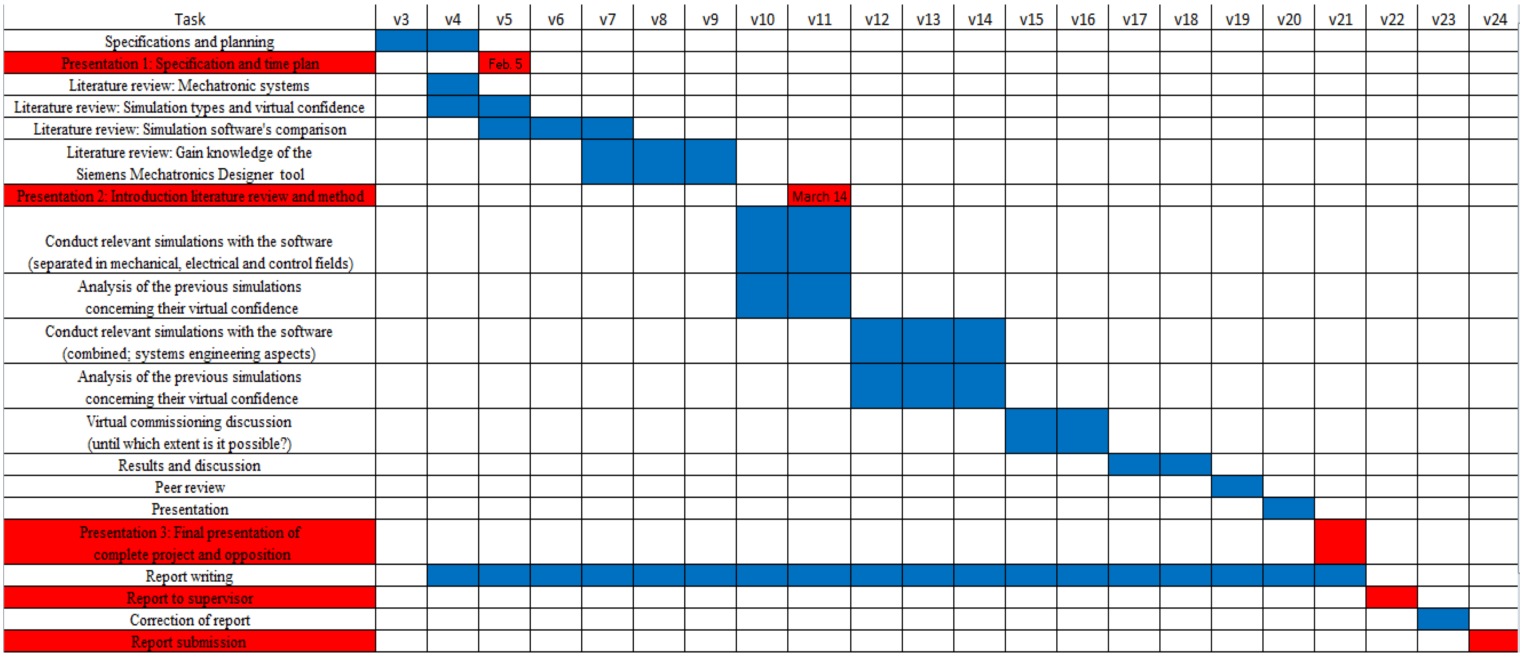

Figure 28: Initial Gantt chart

On the other hand, in the figure 29 it is shown the final project development. Green boxes show the modifications from the initial Gantt.

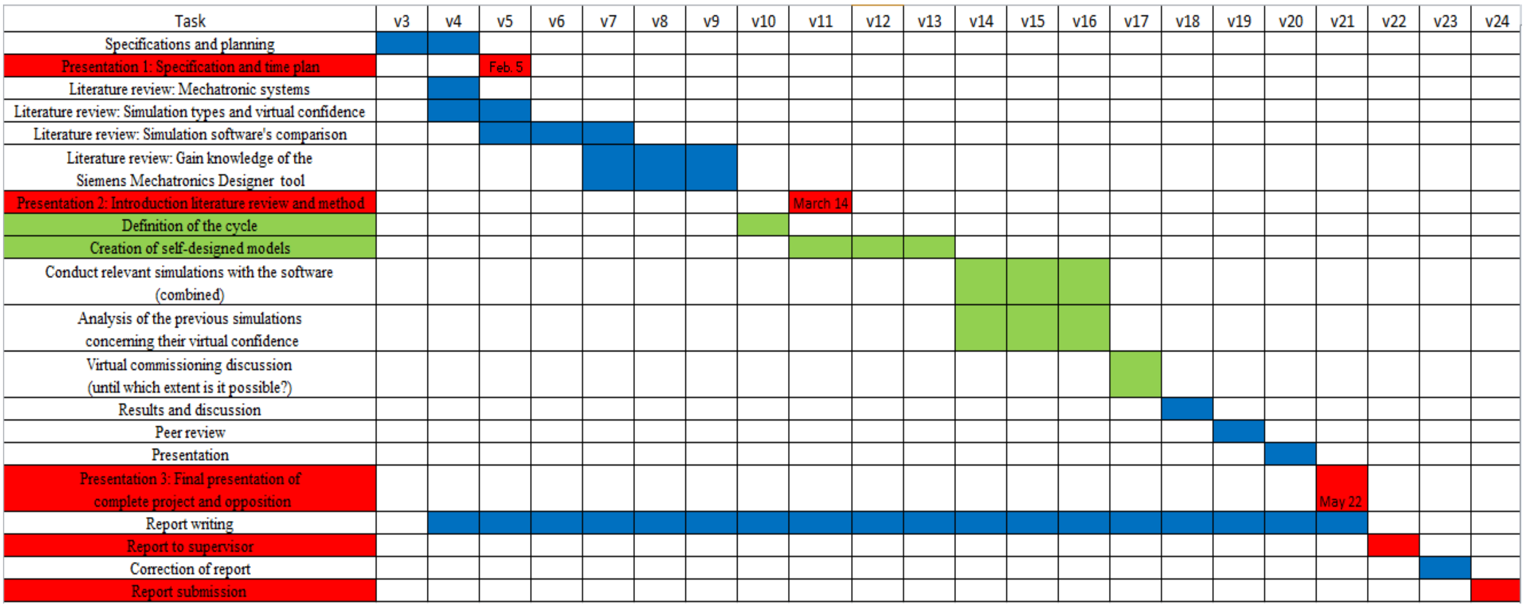

Figure 29: Real project development

The main fact that made the initial plan to vary was that it was not expected to create our own models. However, due to the limitations of the original one, the cycle had to be self-defined and detailed models created. On the other hand, at the beginning it was expected to make the simulations separately on the fields of mechanics, electronics and control, and after that combined. As the software simulates all of these fields together, the individual simulations were not carried out. Finally, the virtual commissioning discussion was delayed from week 16 to week 17, due to unexpected complications on the simulations.